DOCUMENT RESUME

ED 271 971 FL 015 831

AUTHOR McClelland, James L.; Kawamoto, Alan H.
TITLE Parallel Mechanisms of Sentence Processing: Assigning Roles to Constituents of Sentences.
INSTITUTION Carnegie-Mellon Univ., Pittsburgh, Pa. Dept. of Psychology.
SPONS AGENCY Office of Naval Research, Arlington, Va. Personnel and Training Research Programs Office.
PUB DATE 2 May 86
CONTRACT N00014-82-C-0374
NOTE 65p.
PUB TYPE Reports - Research/Technical (143)

EDRS PRICE MF01/PC03 Plus Postage.
DESCRIPTORS *Case (Grammar); Comparative Analysis; Comprehension; Computational Linguistics; Grammar; *Language Processing; Linguistic Theory; *Mathematical Models; Psycholinguistics; *Sentence Structure; Simulation; *Surface Structure
IDENTIFIERS *Word Order

ABSTRACT

This paper describes and illustrates a simulation model for the processing of grammatical elements in a sentence, focusing on one aspect of sentence comprehension: the assignment of the constituent elements of a sentence to the correct thematic case roles. The model addresses questions about sentence processing from a perspective very different from the conventional perspective found in computational linguistics. As a role-assignment model, it consists of two sets of units: one for representing the surface structure of the sentence and one for representing its case structure. Words are treated as patterns of activation; knowledge about them is stored in distributed form in the connections in a large network of simple neuron-like processing units. The model shows facility in dealing with such problems as frame selection, role assignment, and disambiguation, and it suggests a natural way to resolve unappealing aspects of the idea that there is a fixed set of individuated case roles. The simulation model is currently able to process only one-clause sentences, but extensions to multi-clause sentences are considered possible. (Author/MSE)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED271971

Parallel Mechanisms of Sentence Processing: Assigning Roles to Constituents of Sentences

James L. McClelland and Alan H. Kawamoto

Department of Psychology
Carnegie Mellon University

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☐ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

DEPARTMENT
of
PSYCHOLOGY

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

J L McClelland

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

Carnegie-Mellon University

BEST COPY AVAILABLE

Parallel Mechanisms of Sentence Processing: Assigning Roles to Constituents of Sentences

James L. McClelland and Alan H. Kawamoto

Department of Psychology
Carnegie Mellon University

To appear in:

McClelland, J. L., Rumelhart, D. E., and the PDP Research Group, *Parallel distributed processing: Explorations in the microstructure of cognition. Volume II.* Cambridge: Bradford Books, 1986.

This work is supported by ONR Contract N00014-82-C-0374, NR 667-483, by a Research Scientist Career Development Award (MH00385) to the first author, and by grants from the Sloan Foundation and the System Development Foundation. We would like to thank Liz Bates, Gene Charniak, Gary Cottrell, Gary Dell, Mark Derthick, Mark Fanty, Geoff Hinton, Ron Kaplan, Brian MacWhinney, Dave Rumelhart, Bart Selman, and Michael Tanenhaus for useful comments and discussions.

Approved for public release; distribution unlimited.

FL 015831

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution unlimited
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) Technical Report ONR-86-1	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Carnegie-Mellon University 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION Personnel and Training Research Programs Office of Naval Research (Code 1142PT) 800 North Quincy Street
6c. ADDRESS (City, State, and ZIP Code) Department of Psychology Pittsburgh, PA 15213	7b. ADDRESS (City, State, and ZIP Code) Arlington, VA 22217-5000
8a. NAME OF FUNDING / SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER N00014-82-C-0374
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO: 61153N; PROJECT NO: RR04206; TASK NO: RR04206-0A; WORK UNIT ACCESSION NO: NR 667-483
11. TITLE (Include Security Classification) Parallel Mechanisms of Sentence Processing: Assigning Roles to Constituents of Sentences	
12. PERSONAL AUTHOR(S) McClelland, James L., & Kawamoto, Alan H.	
13a. TYPE OF REPORT Technical	13b. TIME COVERED FROM _____ TO _____
14. DATE OF REPORT (Year, Month, Day) April 30, 1986	15. PAGE COUNT
16. SUPPLEMENTARY NOTATION	

17. COSATI CODES		
FIELD	GROUP	SUB-GROUP
05	10	

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
Sentence processing; Language comprehension; Connectionist modeling; Parallel distributed processing

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

OVER

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT ☐ UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL Dr. Harold Hawkins	22b. TELEPHONE (Include Area Code) 202-696-4323 22c. OFFICE SYMBOL 1142PT

DD FORM 1473, 84 MAR 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

ABSTRACT

How do we assign nouns correctly to their underlying case roles in English, and how do we select an appropriate verb frame to assign these nouns to? How do we know whether a noun phrase is a modifier of a preceding noun phrase or an argument of the verb? How do we select the correct meaning of each noun in the sentence, and how do we allow context to modulate its meaning? How do we know how to handle new nouns and verbs? In this article we describe a simulation model that addresses these questions from a perspective quite different from the conventional perspective found in Computational Linguistics. Words are treated as patterns of activation, and knowledge about them is stored in distributed form, in the connections in a large network of simple neuron-like processing units. The model exhibits considerable facility in dealing with the problems of frame selection, role assignment, disambiguation, etc. and suggests a natural way to resolve unappealing aspects of the idea that there is a fixed set of individuated case roles. So far, our simulation model can only process one-clause sentences. Possible extensions to multi-clause sentences are described.

Like many natural cognitive processes, the process of sentence comprehension involves the simultaneous consideration of a large number of different sources of information. In this chapter, we consider one aspect of sentence comprehension: the assignment of the constituents of a sentence to the correct thematic case roles. Case role assignment is not, of course, all there is to comprehension, but it reflects one important aspect of the comprehension process, namely, the specification of who did what to whom.

Case role assignment is not at all a trivial matter either, as we can see by considering some sentences and the case roles we assign to their constituents. We begin with several sentences using the verb *break*:

(1) The boy broke the window.
(2) The rock broke the window.
(3) The window broke.
(4) The boy broke the window with the rock.
(5) The boy broke the window with the curtain.

We can see that the assignment of case roles here is quite complex. The first noun phrase (NP) of the sentence can be the Agent (Sentences 1, 4, and 5), the Instrument (Sentence 2), or the Patient (Sentence 3). The NP in the prepositional phrase (PP) could be the Instrument (Sentence 4), or it could be a Modifier of the second NP, as it is in at least one reading of Sentence 5. Another example again brings out the ambiguity of the role assignment of with-NPs:

(6) The boy ate the pasta with the sauce.
(7) The boy ate the pasta with the fork.

In (6) the with-NP clearly does not specify an Instrument, but in (7) it clearly does.

Before we go much further, it should be said that there is no universally accepted set of case roles, nor universal agreement as to the correct assignment of constituents to roles. We have adopted conventions close to those originally introduced by Fillmore (1968) in "The Case for Case," but we do not think the details are of crucial importance to the behavior of our model. Later we will suggest ways in which an extension of our model might circumvent certain of the difficulties involved in specifying the correct assignment of cases.

These complications aside, it appears from the examples that the meaning of the words in these sentences influences the assignment of arguments to roles. However, the placement of NPs within the sentences is also very important. Consider these two cases:

(8) The vase broke the window.

(9) The window broke the vase.

Here we must rely on word-order constraints. That such constraints are very strong in English can be seen from sentences like:

(10) The pencil kicked the cow.

Even though semantic constraints clearly would indicate that the cow is a much more likely Agent and the pencil a much more likely Patient, Sentence 10 simply is not given this interpretation by adult readers who are native speakers of English.

Word-order constraints like those illustrated by (10) are very strong in English, but it is important to realize that such heavy reliance on such constraints is not universal. Bates and MacWhinney (in press; MacWhinney, Bates, & Kliegl, 1984) have shown that adult speakers of Italian will assign roles to sentences like (10) based predominantly on semantic constraints;[1] word order plays a very limited role and determines assignment only when semantics and case-marking inflections give no information.

As the work of Bates and MacWhinney amply demonstrates, case role assignment is influenced by at least three different kinds of factors: word order, semantic constraints, and (when available) inflectional morphology. Reliance on any one of these constraints is a matter of degree, and varies from language to language. In addition to these factors, there is one more that cannot be ignored, namely, the more global context in which the sentence is presented. Consider, for example, Sentence 11:

(11) The boy saw the girl with the binoculars.

We get one reading if prior context tells us "A boy was looking out the window, trying to see how much he could see with various optical instruments." We get quite a different one if it says "Two girls were trying to identify some birds when a boy came along. One girl had a pair of binoculars and the other did not." Crain and Steedman (1985) have experimentally demonstrated contextual influences on parsing decisions.

While the fact that word order and semantic constraints both influence role assignment has often been acknowledged (Bever, 1970; Fodor, Bever, & Garrett, 1974), there are few existing models that go very far toward proposing a mechanism to account for these effects. However, there are some researchers in language processing who have tried to find ways of bringing semantic considerations into syntactic processing in one way or another. One recent approach has been to rely on the lexicon to influence both syntactic processing and the

---

1. We use the phrase "semantic constraints" to refer to the constraints language users impose on the co-occurrence of constituents in particular roles in case-level representations. In the model, as we shall see, these constraints arise from the co-occurrences of constituents in the experiences the model is exposed to.

construction of underlying functional representations (Ford, Bresnan, & Kaplan, 1982; Kaplan & Bresnan, 1982; MacWhinney & Sokolov, in press). Ford et al. (1982) considered cases like the following:

(12) The woman wanted the dress on the rack.
(13) The woman positioned the dress on the rack.

They noted that the preferred reading of the first of these had *on the rack* as a modifier of *the dress*, while the preferred reading of the second had *on the rack* as a locative argument of *positioned*. To account for this difference in role assignment, they proposed two principles: (a) *lexical preference* and (b) *final arguments*. Basically, lexical preference establishes an expected argument structure (e.g., Subject-Verb-Object in the case of *want*; Subject-Verb-Object-Prepositional Object in the case of *positioned*) by consulting an ordered list of possible argument structures associated with each verb. If a constituent is encountered that could fill a slot in the expected argument structure, the constituent is treated as an argument of the verb. However, if a constituent is encountered that appears to satisfy the conditions on the final argument of the expected argument structure, its attachment is delayed to allow for the incorporation into the constituent of subsequent constituents. Thus, with *want*, the NP *the dress* is a candidate for final argument and is not attached directly as a constituent of the VP; rather, a superordinate NP structure containing *the dress on the rack* is ultimately attached to the VP. With *position*, however, *the dress* would not be the final argument, and so is attached directly to the VP and closed. *On the rack* is then available for attachment as the final argument to the VP.

While this scheme certainly does some of the work that needs to be done in allowing the constraints imposed by the words in a sentence to influence role assignment, we do not think it goes nearly far enough. For as we saw in Sentences 4-7, the NPs of a sentence also influence syntactic decisions. Oden (1978) has verified that all three NPs in sentences like these influence subjects' role-assignment decisions.

In the literature on sentence processing, no one disputes that various factors influence the final reading that is assigned to a sentence. However, there are various views of the way in which these factors are taken into account on-line. Kurtzman (1985) argues that the parsing process is directly guided by an ongoing plausibility analysis; Marslen-Wilson and Tyler (1981) have pioneered this sort of view, and they stress the immediacy with which syntactic, semantic, and pragmatic considerations can all be brought to bear on the course of sentence processing. On the other hand, Frazier and her colleagues (e.g., Frazier & Rayner, 1982; Rayner, Carlson, & Frazier, 1983) argue that the syntactic parser imposes its preferred structuring on the sentence based only on syntactic considerations, passing the results of this processing on quickly to a thematic interpreter that can reject the syntactic parse in favor of a thematically more appropriate reading.

Whichever view one holds, it is clear that a mechanism is needed in which all the constituents of a sentence can work simultaneously to influence the assignment of roles to constituents. While we ourselves tend to favor a highly interactive view, the model we will describe here takes as its input a partial surface parse (though it is one that leaves certain attachment decisions unspecified) and generates from it a case-level representation. Intended extensions of the model, which we will describe below, would incorporate feedback to the syntactic structure level; but most of the model's behavior is not dependent on this feedback, and so readers committed to a less interactive view of the relation between syntactic and thematic analyses may yet find the model to be of interest.

GOALS

The primary goal of our model is to provide a mechanism that can begin to account for the joint role of word order and semantic constraints on role assignment. We wanted the model to be able to *learn* to do this based on experience with sentences and their case representations. We wanted the model to be able to *generalize* what it learned to new sentences made up of novel combinations of words.

In addition, we had several other goals for the model:

- We wanted the model to be able to select contextually appropriate readings of ambiguous words.
- We wanted the model to select the appropriate verb frame based on the pattern of arguments and their semantic features.
- We wanted the model to fill in missing arguments in incomplete sentences with plausible default values.
- We wanted the model to be able to generalize its knowledge of correct role assignment to sentences containing a word it has never seen before, given only a specification of some of the semantic properties of the word.

The model succeeded in meeting all these goals, as we shall see.

The model also exhibits an additional property that we had not actually anticipated, even though it is a central characteristic of language understanding: The model exhibits an uncanny tendency to shade its representation of the constituents of a sentence in ways that are contextually appropriate. It does this without any explicit training to do so; in fact, it does this in spite of the fact that the training inputs it receives are not contextually shaded as they would be in reality. We will examine this aspect of the model's behavior through examples, and observe how it emerges naturally from the model's structure.

The model is, of course, very far from a complete or final model of sentence processing or even case role assignment. Perhaps it is best seen as a partial instantiation of one view of what some properties of the interface between syntactic and more conceptual levels of language representation might be like. We offer the model not because it "solves the problem of sentence comprehension." Rather, we offer it because it suggests new ways of thinking about several aspects of language and language representation. The simulation model that embodies these ideas will undoubtedly require substantial development and elaboration. It is our belief, though, that the basic principles that it embodies will prove extremely valuable as cognitive science continues to try to come to grips with the problem of understanding natural language.

We have limited the model in several ways. Most importantly, we have considered only single clause sentences. We have also considered only a limited set of roles and a limited vocabulary. Since we have restricted the analysis to English, case inflectional morphology does not arise. Within these bounds, we will see that we have been able to meet the goals of the model quite successfully, using a very simple PDP architecture.

Previous, Related Work

Both Cottrell (1985; Cottrell & Small, 1983) and Waltz and Pollack (1985) have preceded us in noting the appeal of connectionism as a means of exploiting the multiple constraints that appear to influence both case role assignment and the contextual disambiguation of ambiguous noun phrases. Their models differ from ours in several ways, most notably in that both rely primarily on local representations (one-unit-one-concept) as opposed to distributed representations, although Waltz and Pollack (1985) do suggest ways that a distributed representation could be used to represent global contextual influences on word meaning disambiguation. Within the context of distributed models, ours builds on the work of J. A. Anderson (1983) and Kawamoto (1985): Both models show how context can be used to select the appropriate reading of an ambiguous word. Our work incorporates mechanisms quite like theirs to accomplish this and other goals. Finally, Hinton's (1981a) early discussion of the use of distributed representations to represent propositions played an important role in the development of the ideas described here.

ARCHITECTURE OF THE MODEL

The role-assignment model is a distributed model, and has many properties in common with the verb learning model described in Chapter 18. The model consists of two sets of units: one for representing the surface structure of the sentence and one for representing its case structure. The model learns through presentations of correct surface-structure/case-structure pairs; during testing, we simply present the surface-structure input and examine the output the model generates at the case-structure level.

Sentences. The sentences processed by the model consist of a verb and from one to three NPs. There is always a Subject NP, and optionally there may be an Object NP. If this is present, there may also be a *with-NP*; that is, a NP in a sentence-final prepositional phrase beginning with the word *with*. All of the numbered sentences considered in the introduction are examples of sentence types that might be presented to the model.

Input format of sentences. What the model actually sees as input is not the raw sentence but a canonical representation of the constituent structure of the sentence, in a form that could be produced by a simple surface parser and a simple lexicon. Such a parser and lexicon are not, in fact, parts of the model in its present form -- the sentences are simply presented to the model in this canonical format. We discuss ways such a parser could be implemented in a PDP model in the discussion section.

Semantic Microfeatures

In the canonical input format, words are represented as lists of semantic microfeatures (Hinton, 1981a; see Chapter 3; Waltz & Pollack, 1985, also make some use of a microfeature representation). For both nouns and verbs, the features are grouped into several dimensions. Each dimension consists of a set of mutually exclusive values, and, in general, each word is represented by a vector in which one and only one value on each dimension is ON for the word and all of the other values are OFF. Values that are set to be ON are represented in the feature vectors as 1s. Values that are set to be OFF are represented as dots (".").

We chose the dimensions and the values on each dimension to capture what we felt were important dimensions of semantic variation in the meanings of words that had implications for the role assignments of the words. We should be very clear about one point, though, which is that we do not want to suggest that the full range of the phenomena that are described under the rubric of the "meanings" of the words are captured by these semantic microfeatures. Indeed, we do not think of words as actually having some fixed meaning at all. Exactly how we do think of meanings will become clear after we examine the behavior of the model, so we postpone a fuller consideration of this issue until the discussion.

The full set of dimensions used in the feature sets are given in Table 1. The noun dimensions are largely self-explanatory, but the different dimensions of the verbs may need some explication. Basically, these dimensions are seen as capturing properties of the scenario specified by the verb. Thus, the DOER dimension indicates whether there is an Agent instigating the event. The CAUSE dimension specifies whether the verb is causal. If not, it indicates whether this is because there is no cause specified (as in the case of *the window broke*) or whether it is because there is no change (as in the case of *the boy touched the girl*). The TOUCH dimension indicates whether the Agent, the Instrument, both, or neither touches the Patient; the "AisP" value simply indicates that the Agent and the Patient are the same (as in *the cat moved*). The NAT_CHNG dimension specifies the nature of the change that takes

Table 1

FEATURE DIMENSIONS AND VALUES

	Nouns
HUMAN	human nonhuman
SOFTNESS	soft hard
GENDER	male female neuter
VOLUME	small medium large
FORM	compact 1-D 2-D 3-D
POINTINESS	pointed rounded
BREAKABILITY	fragile unbreakable
OBJ-TYPE	food toy tool utensil furniture animate nat-inan
	Verbs
DOER	yes no
CAUSE	yes no-cause no-change
TOUCH	agent inst both none AisP
NAT_CHNG	pieces shreds chemical none unused
AGT_MVMT	trans part none NA
PT_MVMT	trans part none NA
INTENSITY	low high

Note: nat-inan = natural inanimate, AisP = Agent is Patient, NA = not applicable.

place in the Patient. The AGT_MVMT and PT_MVMT specify the movement of the Agent and the Patient, respectively; and INTENSITY simply indicates the forcefulness of the action. The labels given to the dimensions are, of course, only for reference; they were chosen so that each noun or verb dimension would have a unique first letter that could be used to designate

the dimension.

It must be stressed that we are not strongly wedded to this particular choice of features, and that other features would need to be included to extend the model to larger sets of nouns and verbs. On the other hand, the features that we did include were carefully chosen because they seemed highly relevant to determining the case role assignments. For example, the DOER dimension directly specifies whether there is or is not an Agent. Thus, the features of the verb, in particular, often have direct case-structural implications. (We would prefer a model that constructed its own semantic microfeatures using back propagation [Chapter 8] or a related method for learning, but this extension has not yet been implemented.)

Figures 1 and 2 give the vectors that we assigned to each of the words used in the model. It will be immediately noted that some of our encoding decisions were arbitrary, and that sometimes we seem to be forcing words into molds that they do not perfectly fit. Further,

	HU	SO	GND	VOL	FORM	PO	BR	OBJ_TYP
ball	1	1	1	1	1	1	1	1
fl-bat	1	1	1	1	1	1	1	1
bb-bat	1	1	1	1	1	1	1	1
bat	1	? ?	? ?	1	? ?	? ?	1	? ?
boy	1	1	1	1	1	1	1	1
paperwt	1	1	1	1	1	1	1	1
cheese	1	1	1	1	1	1	1	1
fl-chicken	1	1	1	1	1	1	1	1
co-chicken	1	1	1	1	1	1	1	1
chicken	1	1	? ?	1	? ?	1	? ?	? ?
curtain	1	1	1	1	1	1	1	1
desk	1	1	1	1	1	1	1	1
doll	1	1	1	1	1	1	1	1
food	1	1	1	1	? ? ? ?	1	1	1
fork	1	1	1	1	1	1	1	1
girl	1	1	1	1	1	1	1	1
hatchet	1	1	1	1	1	1	1	1
hammer	1	1	1	1	1	1	1	1
man	1	1	1	1	1	1	1	1
woman	1	1	1	1	1	1	1	1
plate	1	1	1	1	1	1	1	1
rock	1	1	1	1	1	1	1	1
potato	1	1	1	1	1	1	1	1
pasta	1	1	1	1	1	1	1	1
spoon	1	1	1	1	1	1	1	1
carrot	1	1	1	1	1	1	1	1
vase	1	1	1	1	1	1	1	1
window	1	1	1	1	1	1	1	1
dog	1	1	1	1	1	1	1	1
wolf	1	1	1	1	1	1	1	1
sheep	1	1	1	1	1	1	1	1
lion	1	1	1	1	1	1	1	1

Figure 1. The nouns used in the model and their features. For ambiguous noun constituents, the correct, fully specified reading was used in specifying what the case role representation of the constituent should be, but the underspecified, ambiguous forms were used in the sentence-level input representation. See text for a full discussion.

each feature has the same weight as all the others, and is as definite as all the others. Reality is not nearly so definite or evenhanded, of course. Balls are round, but may be soft or hard; paperweights are generally compact in shape but need not be, etc. The definiteness of the input used in the simulations is a simplification that we have adopted to make the initial coding of the input patterns as straightforward as possible. A more realistic coding would allow some features to be more definite than others. We will see that the model tends to correct this deficiency on its own accord.

One of our goals for the model is to show how it can select the contextually appropriate meaning for an ambiguous word. For ambiguous words (*bat*, flying or baseball, and *chicken*, living or cooked) the input pattern is the average of the feature patterns of each of the two readings of the word. This means that in cases where the two agree on the value of a particular input dimension, that dimension has the agreed value in the input representation. In cases where the two disagree, the feature has a value of .5 (represented by "?") in the input representation. A goal of the simulations is to see if the model can correctly fill in these

	DO	CAU	TOUCH	N_CHG	A_MV	P_MV	IN
ate	1 .	1 . .	. . 1 . .	. . 1 . .	. 1 . .	1 . . .	1 .
ateAVP	1 .	1 . .	. . 1 . .	. . 1 . .	. 1 . .	1 . . .	1 .
ateAVPI	1 .	1 . .	. . 1 . .	. . 1 . .	. 1 . .	1 . . .	1 .
ateAVF	1 .	1 . .	. . 1 . .	. . 1 . .	. 1 . .	1 . . .	1 .
broke	1 .	1 . .	. 1 . . .	1	. 1 . .	. . 1 .	. 1
brokeAVPI	1 .	1 . .	. 1 . . .	1	. 1 . .	. . 1 .	. 1
brokeAVP	1 .	1 . .	1	1	. 1 . .	. . 1 .	. 1
brokeIVP	1 .	. 1 .	. 1 . . .	1	. . . 1	. . 1 .	. 1
brokePV	1 .	. 1 .	. . . 1 .	1	. 1 . .	. . 1 .	. 1
hit	1 .	. . 1	. 1 . . .	. . . 1 .	. 1 . .	. . 1 .	. 1
hitAVPI	1 .	. . 1	. 1 . . .	. . . 1 .	. 1 . .	. . 1 .	. 1
hitAVP	1 .	. . 1	1	. . . 1 .	. 1 . .	. . 1 .	. 1
hitIVP	. 1	. . 1	. 1 . . .	. . . 1 .	. . . 1	. . 1 .	. . 1
moved	1 .	1 . .	1	. . . 1 .	1 . . .	1 . . .	1 .
movedAVP	1 .	1 . .	1	. . . 1 .	1 . . .	1 . . .	1 .
movedAVS	1 .	1 . .	 1	. . . 1 .	1 . . .	1 . . .	1 .
movedPV	. 1	. 1 .	. . . 1 .	. . . 1 .	. . . 1	1 . . .	. 1
touched	1 .	. . 1	. 1 . . .	. . . 1 .	. 1 . .	. . 1 .	1 .
touchedAVPI	1 .	. . 1	. 1 . . .	. . . 1 .	. 1 . .	. . 1 .	1 .
touchedAVP	1 .	. . 1	1	. . . 1 .	. 1 . .	. . 1 .	1 .
touchedIVP	. 1	. . 1	. 1 . . .	. . . 1 .	. . . 1	. . 1 .	1 .

Figure 2. The verbs used in the model and their microfeature representations. The forms followed by strings of uppercase letters (e.g., AVPI) represent the alternative feature patterns that the model must choose between as its way of specifying the contextually appropriate reading of the verb. These alternative feature patterns correspond to the semantic features of the verb appropriate for particular configurations of case roles, as indicated by the uppercase letters: A = Agent, V = Verb, P = Patient, I = Instrument, M = Modifier, S = Self, F = implied Food. The position of the letter indicates the position of the corresponding constituent in the input sentence. The patterns given with the generic verb unadorned by uppercase letters were used in the sentence-level, input representations.

unspecified values, effectively retrieving the contextually appropriate missing values in the process of assigning the word to the appropriate case role. Figure 1 indicates both "full" readings of *bat* and *chicken*, as well as the ambiguous forms used as inputs.[2]

Another goal for the model is to show how it can select the contextually appropriate reading of a verb. This is handled in much the same way as noun ambiguity resolution. The different readings are represented by (potentially) different sets of semantic microfeatures; for example, the Agent/No-Instrument reading of *broke* (brokeAVP) involves contact between the Agent and the Patient, while the Instrument/No-Agent version (brokeIVP) and the Agent/Instrument version (brokeAVPI) involve contact between the Instrument and the Patient. The input representation of the features of a given verb is the same, regardless of context, and the task given to the model is to activate the set of features for the sentence-appropriate version. Rather than use the average pattern based on all of the different possible readings of the verb, we used a "generic" pattern for each verb, which is the pattern for what we took to be the verb's most typical case frame. This is indicated in Figure 2 by the pattern of features next to the plain verb.[3]

The feature patterns corresponding to the different case frames the model must choose among are indicated on the lines in the table following its generic pattern. (The labels on these lines are used simply to designate the feature patterns. They indicate the roles the various arguments in the surface structure of the sentence play. Thus, brokeAVPI specifies the case frame in which the surface subject is the Agent, the surface object is the Patient, and the with-NP is the Instrument.) Note that the microfeatures of two different readings of the same verb may or may not differ, depending on whether the features of the scenario do or do not change in different case frames.

The feature vectors for the constituents of the sentence *The boy broke the window with the hammer* are shown just below the corresponding constituents at the top of Figure 3. Note that these are displayed in the order: Verb, Subject NP, Object NP, and With-NP. The row of letters below the feature vectors indicates the first letter of the name of the dimension on

---

2. For the concept *food*, which is taken to be the implied Patient in sentences like *The boy ate*, no particular shape seems appropriate. Therefore the intended output representation is assumed to be unspecified (as indicated by the "?") for all values on the shape dimension. For all other dimensions, *food* has what we take to be the typical values for foods.

3. The different handling of nouns and verbs is not a principled distinction, but an exploration of two endpoints on a continuum ranging from underspecification of the input for ambiguous words to complete specification of an input representation, regardless of the fact that the features used in the case role representation will differ as a function of context. Perhaps the idea that the features will be altered as a function of context is the best way of putting things in this case. We imagine that the true state of affairs is intermediate between these two extremes, for both nouns and verbs. In any case, the model does not have any prior commitment to the idea that the features in the input representation should be preserved in the output representation; the full prespecification simply gives the model a fuller description to work from, thereby allowing greater differentiation of the different verbs.

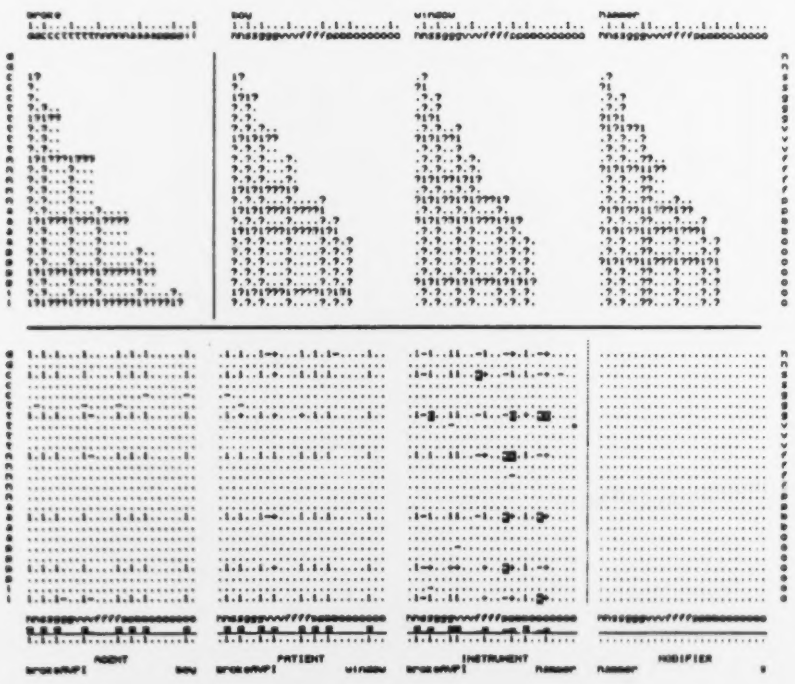

Figure 3. The top line of this figure displays the constituents of the sentence *The boy broke the window with the hammer* in the order: Verb, Subject NP, Object NP, and With-NP. Below these are the microfeatures of each constituent, and below these are conjunctive sentence-structure units for each constituent. Below the horizontal line are the blocks of case-structure units for the Agent, Patient, Instrument, and Modifier roles. Below these is an indication of the pattern of noun features the model is activating for each slot (represented by the vertical black bars), followed by a representation of the microfeatures of the correct filler of each slot. The last line gives the label of the correct head (verb frame or modified NP) and tail (slot filler) for each slot. See text for further explanation.

which each feature represents a value. For example, the first two elements of the verb feature vector are labeled *d* for the DOER dimension; the first two values of each of the three noun feature vectors are labeled *h* for the HUMAN dimension.

Sentence-structure units. The sentence-structure level representation of an input sentence is not actually the set of constituent feature vectors; rather, it is the pattern of activation these vectors produce over units that correspond to *pairs* of features. These units are called

sentence-structure (SS) units.[4]

Each SS unit represents the conjunction of two microfeatures of the filler of a particular surface role. Since there are four sentence-structure roles, there are four sets of SS units. Within each set there is a unit that stands for the conjunction of every microfeature value on each dimension with every microfeature value on every other dimension. For example, for nouns there are units for:

HUMAN = yes / GENDER = male
SOLIDITY = hard / BREAKABILITY = fragile

among many others; for the verb, one of the units corresponds to

DOER = yes / TOUCH = instrument

(i.e., there is a doer -- the Instrument touches the Patient).

The sentence-structure units are displayed in Figure 3 in four roughly triangular arrays. The verb array is separated from the arrays for the three NPs to indicate that different features are conjoined in the verb and NP representations.

Each array contains the conjunctive units for the constituent immediately above it. There is a unit wherever there is a 1, a "?", or a ".". Within each array, the units are laid out in such a way that the column a unit is in indicates one of the microfeatures that it stands for, and the row it is in indicates the other microfeature. Rows and columns are both ordered in the same way as the microfeature vectors at the top of the figure. The dimensions are indicated by the row of letters across the top of each array and along the left (for the verb units) or right (for the three sets of NP units). Note that the set of units in each array fills less than half of each block for two reasons. First, there are only $[n(n-1)]/2$ distinct pairs of n features; second, pairs of values on the same dimension are not included.

We considered various schemes for activating the sentence-structure units. One possible scheme would be to use a strict deterministic activation rule, so that a particular SS unit would be turned on only if both of the features the unit stands for were on in the feature vector. This use of the SS units would allow the model to learn to respond in a finely tuned way to particular conjunctions of microfeatures. However, we wished to see how well the model could function using an inherently noisy input representation. Furthermore, as discussed in

4. An alternative name for these units would be "surface-structure" units, to indicate that they do not capture the notion of underlying subject, object, etc. However, we have chosen the term "sentence-structure" because, for present purposes, the information they capture is not even a full surface-structure parse of the sentence; in particular, it does not specify the attachment of the with-NP.

Chapter 18, we knew that generalization is facilitated when units that only partially match the input have some chance of being activated. In the present case, we considered it important to be able to generalize to words with similar meanings. Therefore, the SS units were treated as stochastic binary units, like the units used in Chapter 18. Each SS unit received excitatory input from each of the two features that it stands for, and we set the bias and variance of the units so that when both of a SS unit's features were active, the unit came on with probability .85; and when neither was active, it came on with probability .15. These cases are represented in the figure by "1" and ".", respectively. Units receiving one excitatory input came on with probability .5; these units are represented in Figure 3 by "?".

The use of the SS units in conjunction with these particular activation assumptions means that the input representation the model must use as the basis for assigning words to case roles is both noisy and redundant. Each feature of the input is represented in the activation of many of the SS units, and no one of these is crucial to the representation. A drawback of these particular activation assumptions, however, is that they do not allow the model to learn to respond to specific conjunctions of inputs. While the model does well in our present simulations, we presume that simulations using a larger lexicon would require greater differentiation of some of the noun and verb representations. To handle such cases, we believe it would be necessary to allow tuning of the input connections to the SS units via back propagation (Chapter 8) so that greater differentiation can be obtained when necessary. In principle, also, higher-order conjunctions of microfeatures might sometimes be required. Our use of broadly tuned, pair-wise conjunctive units illustrates the *style* of representation that we think is appropriate for the input, but the present version is only an approximation to what we would expect a model with a tunable input representation to build for itself.

Case role representation. The case role representation takes a slightly different form than the sentence-structure representation. To understand this representation, it is useful to drop back to a more abstract viewpoint, and consider more generally how we might represent a structural description in a distributed representation. In general, a structural description can be represented by a set of triples of the form (A R B) where A and B correspond to nodes in the structural description, and R stands for the relation between the nodes. For example, a class-inclusion hierarchy can be represented by triples of the form (X IS-A Y), where X and Y are category names. Any other structural description, be it a syntactic constituent structure, a semantic constituent structure, or anything else, can be represented in just this way. Specifically, the case role assignment of the constituents of the sentence *The boy broke the window with the hammer* can be represented as:

Broke Agent Boy
Broke Patient Window
Broke Instrument Hammer

The constituent structure of a sentence such as *The boy ate the pasta with the sauce* would be represented by:

Ate Agent Boy
Ate Patient Pasta
Pasta Modifier Sauce

In a localist representation, we might represent each of these triples by a single unit. Each such unit would then represent the conjunction of a particular head or left-hand side of a triple, a particular relation, and a particular tail or right-hand side. Our more distributed approach is to allocate groups of units to stand for each of the possible relations (or roles), namely, Agent, Patient, Instrument, and Modifier, and to have units within each group stand for conjunctions of microfeatures of the first and third arguments (the head and the tail) of the triple. Thus, the triple is represented not by a single active unit, but by a pattern of activation over a set of units.

In our implementation, there is a group of units for each of the four relations allowed in the case structure. In Figure 3, the Agent, Patient, Instrument, and Modifier groups are laid out from left to right. Within each group, individual units stand for conjunctions of one microfeature of the head of each relation with a microfeature of the tail of each relation. Thus, for example, Broke-Agent-Boy is represented by a pattern of activation over the left-most square block of units. The unit in the ith row and jth column stands for the conjunction of feature i of the verb with feature j of the noun. Thus all the units with the same verb feature are lined up together on the same row, while all the units with the same noun feature are lined up together in the same column. For the Modifier group, the unit in the ith row and jth column stands for the conjunction of feature i of the modified NP and feature j of the modifier NP. Letters indicating the dimension specifications of the units are provided along the side and bottom edges.

The figure indicates the net input to each case role unit produced at the end of the training described below, in response to the sentence *The boy broke the window with the hammer.* (We will see very shortly how these net inputs are produced.) As before, a 1 indicates that the net input would tend to turn the unit on with probability (p) greater than or equal to .85, and a "." indicates that the net input would tend to turn it on with probability of .15 or less. A "+" indicates that the net input has a tendency to turn the unit on ($.85 > p > .5$), and a "-" indicates that the net input has a tendency to turn the unit off ($.5 > p > .15$).

The correct case-frame interpretation of the sentence is provided to the model by a specification that lists, for each of the four possible case roles, the label corresponding to the head and tail of the role. These are shown below each of the four blocks of case role units. The "#" is used to indicate a null slot filler, as in the Modifier role in the present example. From this it is possible to compute which units should be on in the case role representation. Here we simply assume that all the correct conjunctions should be turned on and all other units should be off.

In this example, the pattern of net inputs to the case role units corresponds quite closely to the correct case role representation of the sentence. The features of *boy* may be seen in the columns of the block of Agent units; the features of *window* in the columns of the block of Patient units; and the features of *hammer* in the columns of the block of Instrument units. The features of the Agent-Verb-Patient reading of the verb *broke* can be seen in the rows of each of these three sets of units. There are no features active in the fourth set of units, the Modifier units, because there is no Modifier in this case. In both the Agent and the Patient slots, the model tends to turn on ($p > .5$) all the units that should be on, and tends to turn off ($p < .5$) all the units that should be off. In the Instrument slot, there are some discrepancies; these are indicated by blackening the background for the offending units. All of the discrepancies are relatively mild in that the unit has either a weak tendency to go on when it should not (+ on a black background) or to go off when it should be on (− on a black background).

Several things should be said about the case-frame representations. The first thing is that the slots should not be seen as containing lexical items. Rather, they should be seen as containing patterns that specify some of the semantic properties assigned by the model to the *entities* designated by the words in the sentences. Thus, the pattern of feature values for the verb *break* specifies that in this instance there is contact between the Instrument and the Patient. This would also be the case in a sentence like *The hammer broke the window.* However, in a sentence like *The boy broke the window*, with no Instrument specified, the pattern of feature values specifies contact between the Agent and the Patient. Thus, the verb features provide a partial description of the scenario described by the sentence. The noun features, likewise, provide a partial description of the players (to use Fillmore's analogy) in the scenario, and these descriptions, as we will see later on, may actually be modulated by the model to take on attributes appropriate for the scenario in question.

Details of Sentence Processing and Learning

The model is very much like the verb learning model (Chapter 18). When a sentence is presented, a conventional computer program front-end determines the net input to each of the sentence-structure units, based on the feature vectors of the words. Each of these units is then turned on probabilistically, as described above. Each surface-structure unit has a modifiable connection to each of the case-structure units. In addition, each case-structure unit has a modifiable bias (equivalent to a connection from a special unit that is always on). Based on the sentence-structure pattern and the current values of the weights, a net input to each case-structure unit is computed; this is just the sum of the weights of the active inputs to each unit plus the bias term. Case-structure units take on activation values of 0 and 1, and activation is a probabilistic function of the net input, as in the verb learning model.

During learning, the resulting activation of each case-structure unit is compared to the value it should have in the correct reading of the sentence. The correct reading is supplied as a "teaching input" specifying which of the case role units should be on. The idea is that this

Table 2

GENERATORS FOR SENTENCES USED IN TRAINING AND TESTS

Sentence Frame	Argument Assignment
The human ate.	AVF
The human ate the food.	AVP
The human ate the food with the food.	AVPM
The human ate the food with the utensil.	AVPI
The animal ate.	AVF
The predator ate the prey.	AVP
The human broke the fragile_object.	AVP
The human broke the fragile_object with the breaker.	AVPI
The breaker broke the fragile_object.	IVP
The animal broke the fragile_object.	AVP
The fragile_object broke.	PV
The human hit the thing.	AVP
The human hit the human with the possession.	AVPM
The human hit the thing with the hitter.	AVPI
The hitter hit the thing.	IVP
The human moved.	AVS
The human moved the object.	AVP
The animal moved.	AVS
The object moved.	PV

Note: Argument assignments specify the case role assignment of the constituents of a sentence from left to right. A = Agent, V = Verb, P = Patient, I = Instrument, M = Modifier, F = (implied) Food, S = Self.

teaching input is analogous to the representation a real language learner would construct of the situation in which the sentence might have occurred. Learning simply amounts to adjusting connection strengths to make the output generated by the model correspond more closely to the teaching input. As in the verb learning model, if a unit should be active and it is not, the weights on all the active input lines are incremented and the threshold is decremented. If a unit should not be active but it is, the weights on all the active output lines are decremented and the threshold is incremented. This is, of course, just the perceptron convergence procedure (Rosenblatt, 1962), whose strengths and weaknesses have been examined and relied

upon throughout the book.

SIMULATION EXPERIMENTS

The most important thing about the model is the fact that its response to new inputs is strictly dependent upon its experience. In evaluating its behavior, then, it is important to have a clear understanding of what it has been exposed to during learning. We have done a number of different experiments with the model, but we will focus primarily on one main experiment.

The main experiment consisted of generating a corpus of sentences derived from the sentence frames listed in Table 2. It must be emphasized that these sentence frames were simply used to generate a set of legal sentences. Each frame specifies a verb, a set of roles, and a list of possible fillers of each role. Thus, the sentence frame *The human broke the fragile_object with the breaker* is simply a generator for all the sentences in which *human* is replaced with one of the words on the list of humans in Table 3, *fragile_object* is replaced with one of the words on the list of fragile objects in Table 3, and *breaker* is replaced with one of the words on the list of breakers in the table. It is clear that these generators do not capture all of the subtle distributional properties of referents in real scenarios (e.g., the model is completely sex and age neutral when it comes to hitting and breaking things, contrary to reality), and so we cannot expect the model to capture all these subtleties. However, there are certain distributional facts implicit in the full ensemble of sentences encompassed by the generators. For example, all the breakers but one are hard, not soft (only *ball* is coded as *soft* in the feature patterns); only the humans enter as Agents into scenarios involving Instrument use; etc.

The "target" case-frame representations of the sentences were generated along with the sentences themselves. The case role assignments are indicated in Table 2 by the sequence of capital letters. These indicate the assignment of arguments from the sentences to the roles of Agent, Verb, Patient, Instrument, and Modifier (of the Patient).[5] Note that there are some sentences that could be generated by more than one generator. Thus, *The boy hit the girl with the ball* can be generated by the generator *The human hit the human with the possession*, in which case the ball is treated as a Modifier of the Patient. Alternatively, it may be generated by the generator *The human hit the thing with the hitter*. In this case, the ball is treated as the Instrument. Similarly, *The bat broke the vase* can be generated by *The breaker broke the fragile_object*, in which case its case-frame representation contains a baseball bat serving as Instrument. The same sentence can also be generated by *The animal broke the fragile_object*,

5. Two special cases should be noted: For *The human ate*, the case frame contains a specification (F) that designates an implied Patient that is the generic food with unspecified shape, as indicated in the feature patterns displayed in Figure 1. For *The human moved* and *The animal moved*, the case frame contains a specification (S) that indicates that there is an implied Patient who is the same as the Agent (note that the sense of the verb *move* used here involves moving oneself and not one's possessions).

in which case, of course, its case-frame representation contains a flying bat serving as Agent.

For the main experiment, we generated all the sentences covered by the generators and then selected eight of each type to use as training sentences. Of these we selected two to be *familiar* test sentences. In addition, we selected two additional sentences from each generator to be used as *novel* test sentences. These sentences were never used to train the model.[6]

The model was given 50 cycles of training with the set of training sentences. On each cycle, each sentence was presented, the model's response to it was generated, and connection strengths were adjusted according to the perceptron convergence procedure.

Table 3

NOUN CATEGORIES

human	man woman boy girl
animal	fl-bat li-chicken dog wolf sheep lion
object	ball bb-bat paperwt cheese co-chicken curtain desk doll fork hatchet hammer plate rock pasta spoon carrot vase window
thing	human animal object
predator	wolf lion
prey	li-chicken sheep
food	co-chicken cheese spaghetti carrot
utensil	fork spoon
fragile_object	plate window vase
hitter	ball bb-bat paperwt hatchet hammer rock vase
breaker	paperwt ball bb-bat hatchet hammer rock
possession	ball dog bb-bat doll hatchet hammer vase

6. Some of the generators (e.g., *The human hit the thing with the hitter*) generate rather large numbers of different sentences (in this case, 756), but others (e.g., *The human ate, The predator ate the prey*) generate only very small numbers of sentences (four in each of these cases). The training materials contained four copies of each of two of these sentences so that even here, there were two familiar test sentences and two unfamiliar test sentences.

After the 5th, 10th, 20th, 30th, 40th, and 50th training cycles, the model was tested on both the familiar and the novel test sentences. No learning occurred during the tests, so that the response to the novel test sentences always represented generalization from the training materials rather than the effects of direct experience.

Basic Results

Figure 4 gives a very global indication of the model's performance on both the familiar and the novel test sentences at each phase of testing. The figure indicates the average number of incorrect microfeatures produced by the model as a function of learning trials, for both the familiar and unfamiliar test sentences. There are a total of 2500 case role units, so on the average the model is getting over 95% correct, even with unfamiliar sentences, after the first 5 learning cycles, and is down to about 1% error at Cycle 50. However, these statistics are somewhat misleading, since on the whole, most of the case role units should be off. A more realistic indication of the absolute performance level is provided by the observation that between 56 and 168 of the units should be on in the correct case role representation of each sentence. In general, the errors that the model does make are about evenly distributed between sins of commission (false alarms) and sins of omission (incorrect rejections). Thus, on average, at the end of 50 cycles, the model is turning on about 85 of the approximately 100 microfeatures that should be on, and is turning on about 15 out of 2400 microfeatures that should be off. This corresponds to a d of about 3.5.

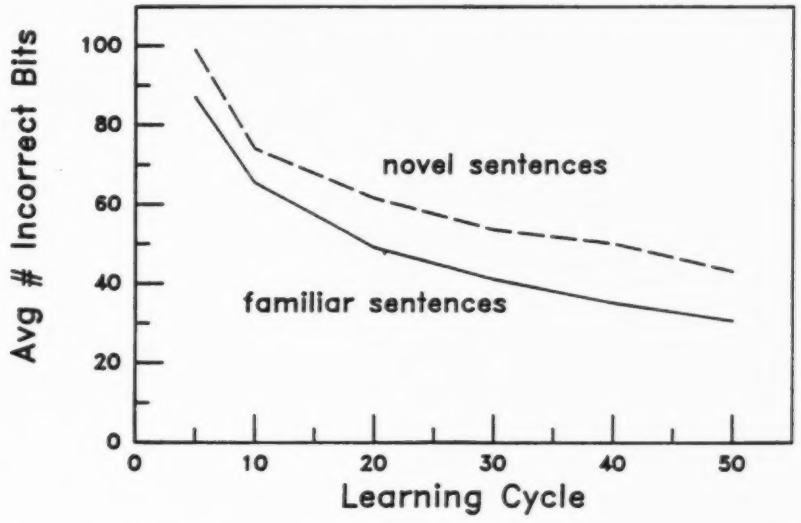

Figure 4. Average number of incorrect microfeatures produced as a function of amount of learning experience (in number of cycles through the full list of training sentences).

Two things are apparent in the graph. First, there is a smooth and continuing improvement in performance for both familiar and unfamiliar sentences. Second, there is an advantage for sentences that the model has actually seen before, but it is not particularly great. As we shall see, there is considerable variability in the model's ability to deal with particular sentences; only a part of it is accounted for by whether a particular sentence happens to be old or new.

Figures 5 and 6 indicate the model's performance on each of the 38 familiar and unfamiliar test sentences, at each of the six tests.

FAMILIAR SENTENCES

					Number of Cycles of Training					
	Input			Frame	5	10	20	30	40	50
man	ate			AVP	72	38	23	30	20	10
girl	ate			AVP	54	36	23	11	13	9
woman	ate	cheese		AVP	49	32	26	15	12	27
woman	ate	pasta		AVP	62	33	22	14	11	9
woman	ate	co_chic	pasta	AVPW	90	66	49	37	38	29
man	ate	pasta	co_chic	AVPW	78	47	31	27	25	10
girl	ate	pasta	spoon	AVPI	72	63	55	26	19	23
boy	ate	co_chic	fork	AVPI	73	47	45	20	26	8
dog	ate			AVP	58	27	13	22	26	12
sheep	ate			AVP	75	44	25	19	15	19
lion	ate	ll_chic		AVP	60	48	31	17	19	10
lion	ate	sheep		AVP	52	34	11	21	10	9
woman	broke	window		AVP	65	38	33	21	14	13
boy	broke	plate		AVP	59	56	17	31	20	18
man	broke	window	bb_bat	AVPI	86	70	69	36	47	25
boy	broke	plate	hatchet	AVPI	101	52	37	35	30	14
paperwt	broke	vase		IVP	76	60	47	42	40	35
bb_bat	broke	plate		IVP*	108	91	94	111	83	47
fl_bat	broke	window		AVP*	107	149	119	105	111	108
wolf	broke	plate		AVP	91	62	44	47	26	30
vase	broke			PV	70	55	33	31	14	21
window	broke			PV	73	41	24	27	16	18
man	hit	pasta		AVP	85	55	34	30	19	16
girl	hit	boy		AVP	77	76	38	41	23	33
man	hit	girl	hatchet	AVPW*	138	126	95	83	60	83
man	hit	woman	hammer	AVPW*	130	112	79	63	59	69
woman	hit	fl_bat	hammer	AVPI*	114	67	87	46	41	33
girl	hit	vase	bb_bat	AVPI	147	102	81	56	54	49
hatchet	hit	pasta		IVP	100	70	49	41	34	24
hammer	hit	vase		IVP	103	75	57	55	49	32
man	moved			AVS	83	77	41	35	36	20
woman	moved			AVS	104	68	33	40	24	37
woman	moved	plate		AVP	79	64	49	32	36	31
girl	moved	pasta		AVP	66	57	45	29	29	18
fl_bat	moved			AVS*	122	109	103	71	50	52
dog	moved			AVS	97	64	46	64	45	42
doll	moved			PV	148	103	82	84	90	69
desk	moved			PV	93	86	48	44	50	42

Figure 5. Number of microfeature errors for each familiar sentence, after 5, 10, 20, 30, 40, and 50 cycles through the set of training stimuli. Sentences are written in SVOW (W = with-NP) order to facilitate reading; the column labeled "Verb Frame" indicates the role assignments of these arguments. The * indicates a sentence that is ambiguous in terms of the set of generators used, in that it could have been generated in two different ways. Recall that ambiguous words (*bat*, *chicken*) are ambiguously specified in the input; it is the job of the model to select the contextually appropriate reading.

In what follows, we will focus attention on the performance of the model at the end of 50 learning cycles.

Use of Semantic and Word-Order Cues

To assess the model's ability to make use of word-order and semantic cues to role assignment, we examined its performance on the verbs *hit* and *break*. In the case of *break*, constraints are very important; in the case of *hit*, role assignment is sometimes determined by word order alone.

The dog broke the plate, the hammer broke the vase, and the window broke. The verb *break* can take its Instrument, its Agent, or even its Patient as its Subject. In the first two cases (as in *The dog broke the plate* and *The hammer broke the vase*), it is only the semantic properties of the subject that can tell us whether it is Agent or Instrument.

UNFAMILIAR SENTENCES

					Number of Cycles of Training					
	Input			Frame	5	10	20	30	40	50
boy	ate			AV	87	28	34	19	18	8
woman	ate			AV	80	41	28	18	19	18
woman	ate	co_chic		AVP	64	43	31	25	23	19
man	ate	co_chic		AVP	51	49	34	29	21	21
woman	ate	co_chic	carrot	AVPN	121	98	101	70	60	64
boy	ate	carrot	pasta	AVPN	116	86	83	69	64	59
man	ate	co_chic	fork	AVPI	86	66	39	43	32	15
woman	ate	carrot	fork	AVPI	89	68	44	40	32	28
fl_bat	ate			AVP	79	81	43	38	45	43
ll_chic	ate			AVP	104	57	62	56	39	46
wolf	ate	sheep		AVP	58	45	44	24	18	14
wolf	ate	ll_chic		AVP	63	45	32	27	34	29
fl_bat	broke	vase		AVP*	113	128	133	109	119	136
dog	broke	plate		AVP	68	52	35	28	28	23
girl	broke	plate		AVP	82	52	21	16	14	14
woman	broke	plate		AVP	81	45	27	30	26	15
girl	broke	vase	hatchet	AVPI	96	67	55	32	24	28
man	broke	vase	ball	AVPI	147	91	80	74	58	60
hammer	broke	vase		IVP	86	61	55	44	62	32
ball	broke	vase		IVP	115	83	80	85	72	57
plate	broke			PV	73	60	34	36	29	26
plate	broke			PV	62	47	35	27	23	25
boy	hit	girl		AVP	84	62	31	44	31	36
girl	hit	carrot		AVP	86	68	45	42	35	34
man	hit	boy	hammer	AVPM*	159	126	121	101	96	71
boy	hit	woman	doll	AVPM	127	127	121	93	96	80
girl	hit	spoon	rock	AVPI	152	112	99	94	94	73
girl	hit	curtain	ball	AVPI	178	143	112	117	86	94
paperwt	hit	co_chic		IVP*	115	81	71	74	59	48
rock	hit	plate		IVP	117	97	65	58	58	57
girl	moved			AVS	115	95	73	70	74	57
boy	moved			AVS	101	84	67	53	55	50
girl	moved	hammer		AVP	103	82	56	46	43	42
man	moved	window		AVP	95	78	77	58	58	43
wolf	moved			AVS	101	74	61	60	54	35
sheep	moved			AVS	117	108	79	77	60	64
hatchet	moved			PV	95	76	60	60	67	40
paperwt	moved			PV	101	70	79	59	68	45

Figure 6. Number of missed features in output for each unfamiliar sentence, after 5, 10, 20, 30, 40, and 50 cycles of training. See previous caption for conventions.

These two sentences happened to be among the unfamiliar test sentences generated by the generators *The animal broke the fragile_object* and *The breaker broke the fragile_object.*[7] The patterns of activity produced by the model in response to each of these two sentences are shown in Figures 7 and 8. We can see that in the case of *The dog broke the plate*, there is a strong pattern of activation over the Agent units, while in the case of *The hammer broke the vase*, there is a strong pattern of activation over the Instrument units. In fact, the pattern of activation over the Agent units in the first case corresponds closely to the expected target pattern for brokeAVP-agent-dog, and the pattern of activation over the Patient units in the second case corresponds closely to the expected target pattern for brokeIVP-instrument-hammer. Thus, the model has correctly assigned the word *dog* to be Agent in the first case,

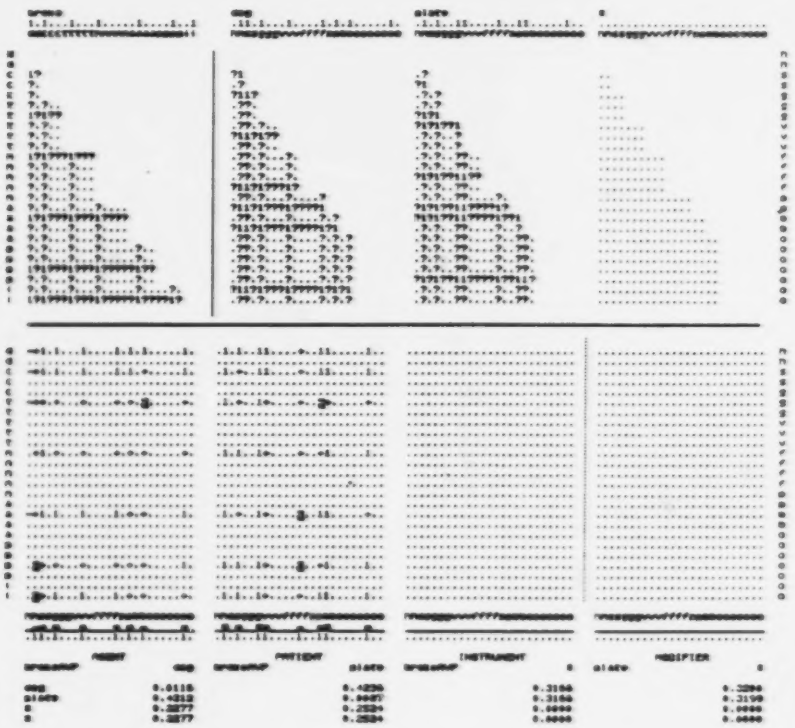

Figure 7. Display of the state of the model after processing the sentence *The dog broke the plate.*

7. The others were *The bat broke the vase* and *The ball broke the vase.* Each of these sentences will be discussed later.

and the word *hammer* to be Instrument in the second. A summary display that makes the same point is shown below the case role representations in Figures 7 and 8, and is repeated, for these and other sentences, in Figure 9.

The summary display indicates in histogram form the features of the right-hand side of the triple stored in the corresponding block of units. The summary pattern for the Agent role in the sentence *The dog broke the plate* is shown below the block of Agent units in Figure 7. The pattern just below this indicates the "correct" pattern; in this instance, this pattern is just the feature pattern for *dog*. In this case we can see that the pattern the model produced is very close to the correct answer.[8]

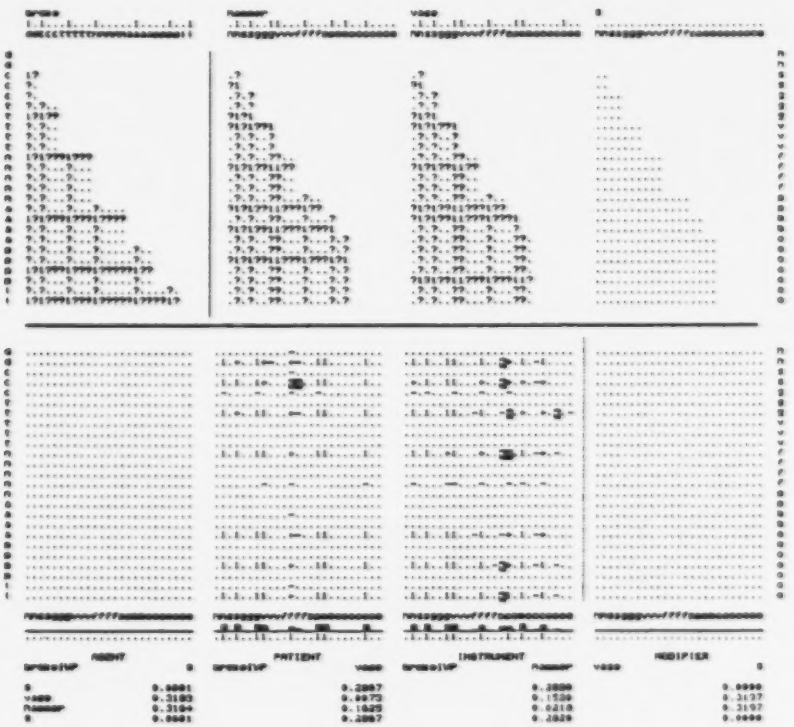

Figure 8. Display of the state of the model after processing the sentence *The hammer broke the vase.*

8. The summaries are based on average probability of activation, with the average only taken over the subset of units associated with a particular noun feature that is also associated with the features of the correct verb frame. Thus, for example, there are 25 units for the feature *nonhuman* in each of the four case slots. These units are displayed as the second column of units in each slot; the ones that indicate whether this feature holds for the Agent are shown in the second column of the Agent units, the left-most block. Seven of these units conjoin this feature with a feature of

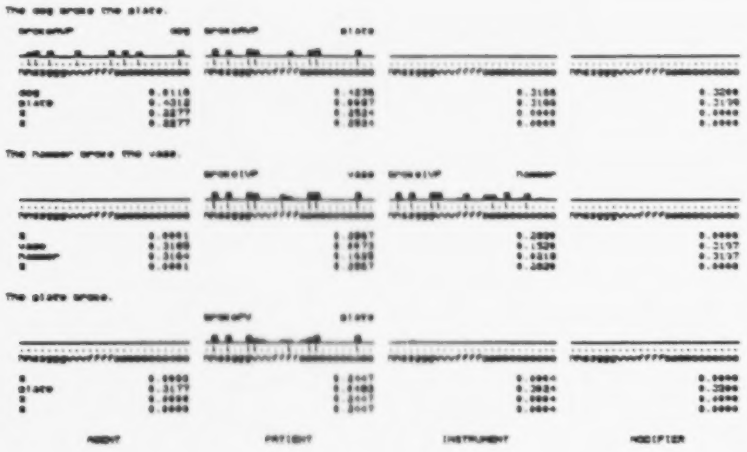

Figure 9. Summary of the activations produced on the case role units by the sentences *The dog broke the plate*, *The hammer broke the vase*, and *The plate broke*.

Below each feature-pattern summary, we present in quantitative form a measure of the similarity of all possible nouns from the sentence with the summary pattern for each slot. We also compare the contents of each slot to the "#" pattern, corresponding to "all units off." The numbers (called *deviation scores*) indicate the mean squared deviation of the indicated noun pattern from the summary representation of the contents of each slot. A small deviation score indicates a close correspondence. Thus, if we look at the row of numbers labeled *dog*, we can see again that the pattern over the Agent units (first column) is very similar to the pattern for *dog*. Further, we can see that the pattern for *dog* is not similar to the patterns in any of the other slots (second through fourth columns). The pattern on the Patient units is similar to the pattern for *plate*, and the patterns on the Instrument and Modifier units are similar to the null pattern. For the sentence *The hammer broke the vase*, the pattern on the Instrument units is similar to the pattern for *hammer*, the pattern on the Patient units is similar to the pattern for *vase*, and the patterns on the Agent and Modifier units are similar to the blank pattern. Thus, each argument has effectively been assigned the correct role, and each role has effectively been assigned the correct filler.[9]

---

the verb frame brokeAVP. In the case of *The dog broke the plate*, illustrated in Figure 7, most of these units have a tendency to come on ($.5 < p < .85$), while two (the last two) have a tendency to stay off ($.15 < p < .5$) -- a tendency that is considered to be erroneous and results in these units having black backgrounds. The average of these probabilities is about .65. This value is translated into the height of the bar in the second position of the Agent slot for the sentence *The dog broke the plate*.

9. With respect to the summary pattern on the Instrument slot for *The hammer broke the window*, we can see that the model is trying to activate both values on the POINTINESS dimension. Because *hammer* and *hatchet* are ident-

This example has been described in some detail, in part to explicate our displays of the model's responses to particular sentences. In the process we have seen clearly that *dog* and *hammer* trigger the appropriate bindings of arguments to slots. This is also the case for examples of the form *The plate broke*. There, Figure 9 indicates that the only slot in which there is appreciable activity is the Patient slot.

These examples illustrate that the model has learned quite a bit about assigning fillers to slots on the basis of the microfeatures of the slot fillers involved. For the word *break*, animate surface subjects are treated as Agents and inanimate surface subjects are treated as Instruments if an Object is specified; if not, the inanimate surface subject is treated as Patient. The model seems to capture this fact pretty well in its behavior.[10]

The boy hit the girl and the girl hit the boy. For the verb *hit*, there is a possibility that a sentence describing an instance of hitting will have two animate arguments, which may be equally plausible candidates to serve as Agent. The only way to tell which is the Agent (in the absence of other context) is to rely on word-order information. We know that *boy* is the Agent in the *The boy hit the girl* only because it occurs in the preverbal position. The model has no difficulty coping with this fact. Figure 10 shows in summary form the features activated by the sentences *The girl hit the boy* (a sentence the model actually experienced during learning) and *The boy hit the girl* (a sentence not experienced during learning). In both cases, the model activates the feature pattern for the correct argument in the correct slot. This is so, even though the feature patterns for *boy* and *girl* differ by a single feature.[11] As a more formal test of the model's ability to assign slot fillers to slots based only on sentence-structure information, we tested the model on the full set of 12 different *human-hit-human* sentences. In all cases, the preverbal argument more closely matched the pattern of activation in the Agent

---

ical except for this feature, the representation is really describing a somewhat pointy hammer (or perhaps blunt hatchet).

10. One thing that we see illustrated here is that with certain novel sentences, the model may have a tendency to misgenerate some of their features when assigning them to underlying slots. Thus, for example, in the case of *The plate broke*, only some of the features are produced faithfully. These are, in fact, the features associated with the slot fillers that the model actually learned to deal with in this sentence frame (*vase*, *window*). The ones that are poorly reproduced are the features that the familiar exemplars differ on or which differ between the familiar examples and *plate*. Such errors would be greatly reduced if a more disparate range of Patient-intransitive verbs with a more disparate range of subjects had been used in the learning. Such errors could also be cleaned up quite a bit by an auto-associative network of connections among the case role units. The virtues of augmenting the model with such a network are considered in more detail later.

11. Though the model did not actually learn the sentence *The boy hit the girl* or any other sentence containing *boy* and *girl* as Subject and Object, it did learn several sentences in which *boy* was the subject of *hit* and several others in which *girl* was the object. As it happened, several of these involved modifiers of *girl*, hence the rather diffuse pattern of activation over the Modifier units.
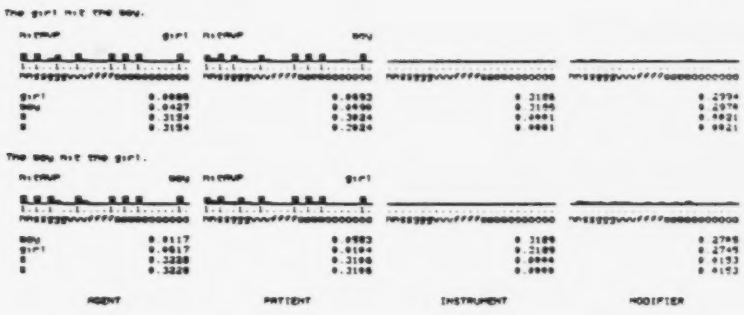
Figure 10. Summaries of of the role assignment patterns produced by *The girl hit the boy* and *The boy hit the girl*.

role, and the postverbal argument more closely matched the pattern in the Patient role.

Verb-Frame Selection

Part of the task of the model is to put slot fillers in the correct places, but there is more that it must do than this. It must also determine from the ensemble of arguments what reading of the verb is intended. By this, we mean which of the possible scenarios the verb might describe is actually being described in this particular case. For example, the sentences *The boy broke the window with the hammer* generates quite a different mental scenario than *The dog broke the window* or *The window broke*. Our model captures the differences between these scenarios in two ways: one is simply in terms of the set of underlying roles and the assignment of sentence constituents to these roles. We have already seen in our earlier discussion of *break* that the model produced a different set of slots and assigned the preverbal noun phrase to a different role in each of the sentences *The dog broke the window*, *The hammer broke the vase*, and *The plate broke*. The other way the model captures the differences between scenarios is in terms of the pattern of activation of verb features in the Agent, Patient, and Instrument slots. Thus in *The boy hit the girl*, we visualize physical contact between the boy and the girl; in the case of *The boy hit the girl with the rock*, we visualize physical contact between the rock and the girl. These and other related distinctions are captured (admittedly, imperfectly) in the different feature patterns associated with the verb frames. As with the feature patterns for nouns, the patterns that we used do not adequately encode the differential flexibility of different aspects of the different scenarios. Nevertheless they capture the essence of the difference, say, between one or the other kind of hitting.

The features of the scenario are captured in the pattern of activation of the case role units. To this point, we have been summing along columns of units to determine the features of the object assigned to a particular role by the model. To determine the features of the

action or scenario assigned to the sentence, we need to look along the rows. Figures 3, 7, and 8 indicate that features in somewhat different patterns of rows are activated by the sentences *The boy broke the window with the hammer*, *The dog broke the plate*, and *The hammer broke the vase*. These are, indeed, the correct verb features in each one. Thus, we see that the model has learned to successfully assign a different scenario, depending on the arguments supplied along with the verb.

In general, the model did quite well selecting the correct scenario. For every unambiguous test sentence, familiar or unfamiliar, the value on each scenario dimension that was most active was the correct value.

Filling in Missing Arguments

The model does a very good job filling in plausible default values for missing arguments. To demonstrate this, we tested the model on the sentence fragment *The boy broke*. The results are shown in Figure 11. The model fills in, fairly strongly, a plausible but underspecified fragile object -- something that is nonhuman, neuter, nonpointed, fragile, and has object-type *furniture* (*plate*, *vase*, and *window* are all classified as furniture in the model). Values on the size (VOLUME) and shape (FORM) dimensions are very weakly specified.[12]

We see similar kinds of things happening with *The girl ate*, though in this case, the model is actually taught to fill in *food* of unspecified form, so this performance is not surprising. Something slightly different happens with *The man moved*: The model is taught to treat *man* as both Agent and Patient in such sentences; and indeed, the pattern for *man* predominates in the Patient slot. (Note that if an object is specified, as in *The man moved the piano*, it is handled correctly.)

These additional examples make two points: First, that the model can be explicitly trained to fill in implied arguments; and second, that the filling in of implied arguments is clearly specific to the particular verb. What is filled in on the Patient slot is quite different for each of these three examples.

---

12. While the model's response to *The boy broke* clearly illustrates default assignment, it differs from the way many people appear to process this input; several people have commented to us that they read this fragment as a complete sentence specifying that it was the boy that broke, even though this produces a somewhat anomalous reading. The model's response to *The boy broke* is closer to the kind of thing most people get with a verb like *hit*, which does not have an intransitive reading.

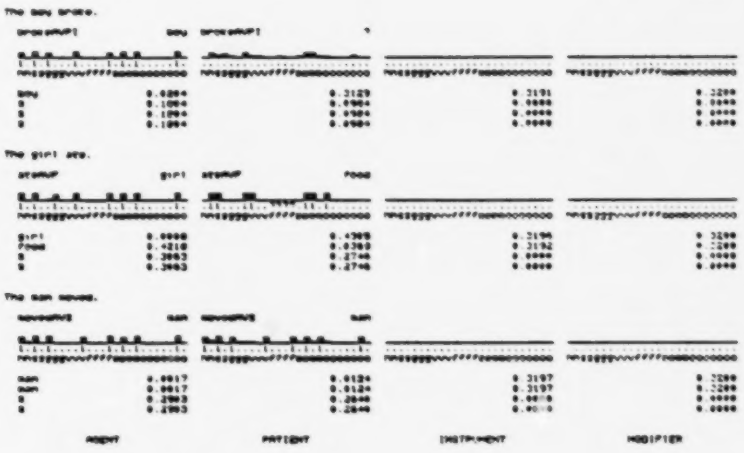

Figure 11. Summaries of the model's response to *The boy broke*, *The girl ate*, and *The man moved*.

Lexical Ambiguity Resolution

In general, the model does very well with ambiguous words. That is, it has little difficulty determining which reading to assign to an ambiguous word based on its context of occurrence -- as long as the context is itself sufficient to disambiguate the meaning of the word.

To demonstrate this point, we carried out a number of analyses of the model's responses to sentences containing the ambiguous nouns *chicken* (live or cooked) and *bat* (flying or baseball). We divided the sentences containing these ambiguous words into those that had only one case-frame representation derivable from the generators and those that could have been generated in two different ways. An example of the former kind of sentence is *The chicken ate the carrot*, since only a live chicken could occur as the preverbal NP with *ate*. An example of the latter kind of sentence is *The bat broke the window*, which could be generated either from *The animal broke the fragile_object* or from *The breaker broke the fragile_object*. Given the set of generators, the context specifies which reading is correct for the first kind of sentence but does not specify which reading is correct for the latter. Since our present interest is to see how well the model can do at using context to resolve ambiguity, we focus here on the former type.

Our first analysis simply compared the model's performance with these sentences to its performance with sentences containing no ambiguous words. Since a large number of factors contribute to the number-of-bits-incorrect measure, we focused on a set of seven matched pairs of sentences that were generated from the same generators and were also either both old

or both new, but differed in that one contained an ambiguous word (e.g., *The woman ate the chicken with the carrot*) and the other did not (*The boy ate the carrot with the pasta*). The average number of features missed was 28.4 for the items containing ambiguous words and 29.1 for the control items, $F(1,6) < 1$.

Another way to examine the data is to examine the relative strengths of the features of the two readings of an ambiguous word in the output summary representation. Figure 12 indicates a typical case and one interesting exception to the general pattern. In the typical case (*The man ate the chicken with the fork*), we see that the features of the cooked chicken have been strongly activated in the Patient slot, and though there are some weak traces of the features of the live chicken, they are not stronger than the weak extraneous activation we often see of incorrect features for unambiguous words. In the atypical case (*The wolf ate the chicken*), the pattern is really much closer to the cooked chicken than the live one that the model was "supposed" to have retrieved. It is difficult to fault the model too much in this case, however. Though it was never given sentences of the form *The animal ate the food*, it was given sentences of the form *The animal ate*, where the underlying case frame included (implied) *food*. Thus the model had considerable reason to treat the object of *The animal ate* as *food*. Though it had learned *The lion ate chicken* referred to a live chicken, it appears to prefer to treat *chicken* in *The wolf ate chicken* more as cooked food than as a living animal. Some of the properties of the live chicken are weakly present -- e.g., its female sex and its natural, animate object-type classification -- but the predominant pattern is that of *food*.

With this exception, the model has shown itself quite capable of handling sentences containing ambiguous words as well as sentences containing only unambiguous words. This is, of course, not really surprising in view of the fact that all the words in the sentence are used to help constrain the features assigned to the fillers of every role in the case frame. When one word does not provide the information to constrain the output, the model can exploit

Figure 12. Summaries of the case role activations produced by *The man ate the chicken with the fork* and *The wolf ate the chicken.*

information contained in other words.

Structural Ambiguity

In this section, we briefly consider another type of ambiguity that sometimes arises even when the words in the sentence are unambiguous. This is the structural ambiguity of sentences such as *The man hit the woman with the hammer*. In such cases, *hammer* can either be the Instrument or simply a Modifier of *woman*. This case-structure ambiguity parallels an ambiguity in the syntactic structure of the sentence as well; if *hammer* is an Instrument, it is dominated directly by the VP; whereas if it is a Modifier, it is dominated by the NP *the window*. Because *hammer* was included both in the list of possible possession-modifiers of human objects and in the list of possible instruments of hitting (designated as *possessions* and *hitters* in the sentence frames in Table 2 and in the noun categories in Table 3), either of these readings is equally possible. Thus, it is not so surprising that the model has considerable difficulty with such sentences, generating a blend of the two readings.

A particularly interesting case of case-structure ambiguity occurs with the sentence *The bat broke the window*. As already mentioned, the sentence is both lexically ambiguous and ambiguous in case structure, and the structural ambiguity hinges on the lexical ambiguity. If *bat* is a baseball bat then the case frame specifies an Instrument and a Patient, but if it is a flying bat then the case frame specifies an Agent and a Patient.

Figure 13 illustrates the pattern of activation generated in this case. What the model does, quite sensibly, is activate one kind of *bat* -- the flying bat -- on the Agent units and the other -- the baseball bat -- on the Instrument units (see the figure caption for a detailed explanation of the use of the black background in this figure).

People generally get only one reading of a sentence at a time, even when (as in this case) the sentence is easy to interpret in either of two ways. In a later section of this chapter we explain how cross-connections among the case role units and back-connections to sentence-level units would tend to cause the model to choose a single interpretation, even in these ambiguous cases.

Shades of Meaning

Another property of the model, related to its handling of ambiguity, is the fact that it can shade the feature patterns it assigns, in ways that often seem quite appropriate. One example of this tendency arises in the model's response to the sentence *The ball broke the vase*. A summary of the pattern produced is shown in Figure 14. The feature pattern on the Instrument units matches the features of *ball* fairly closely. The value on each dimension that is most strongly activated is the correct value, except on one dimension -- the hard/soft dimension. On this dimension, we can see that the model has gotten *ball* completely wrong -- it has strongly activated *hard*, instead of *soft*.

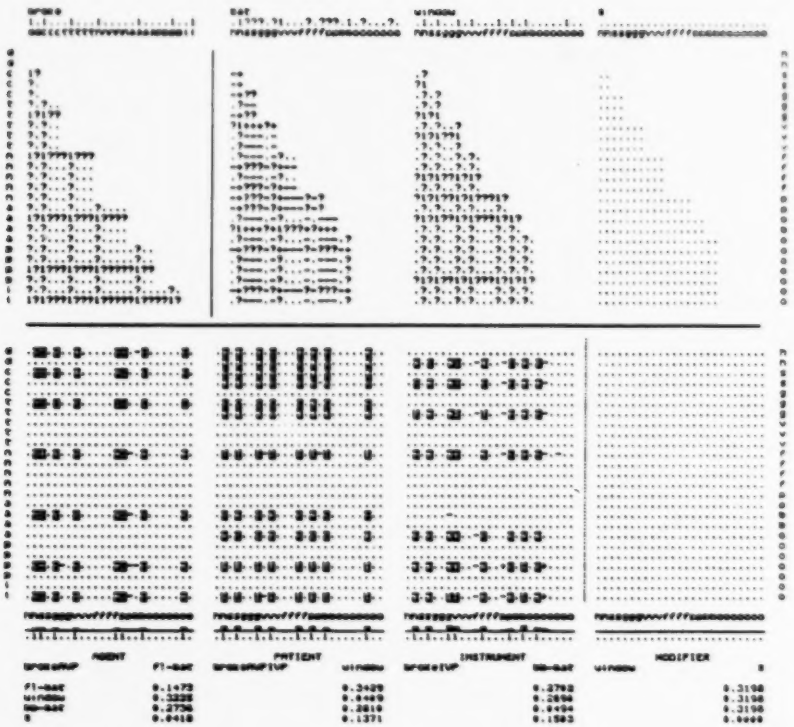

Figure 13. State of the model after processing the sentence *The bat broke the window.* For this sentence, the black background on the Agent units shows which units would be active in the pattern for *flying bat* as Agent of the agentive (AVP) reading of *broke.* The black background on the Instrument units indicates the units that would be active in the pattern for *baseball bat* as Instrument of the instrumental (IVP) reading of *broke.* The black background on the Patient units indicates the units that would be active in *either* the pattern appropriate for *window* as Patient of the agentive reading of *broke or* in the instrumental reading of *broke.*

Figure 14. Summary of the activations on the case role units for the sentence *The ball broke the vase.*

In one sense, this is clearly an "error" on the model's part; all the balls that it learned about were soft, not hard balls. But in another sense, it is a perfectly sensible response for the model to make. All of the other instruments of breaking (called *breakers* in Table 3) were, in fact, hard. The model picked up on this fact, and shaded its interpretation of the meaning of the word *ball* as a result. As far as this model is concerned, balls that are used for breaking are hard, not soft.

This kind of shading of meaning is just another manifestation of the process that fills in missing arguments, chooses appropriate verb frames, and selects contextually appropriate meanings of words. It is part and parcel of the mechanism that generally results in the activation of the nominally correct feature pattern. It is a mechanism that naturally blends together what it learns into a representation that regularizes slot fillers, in the same way that the verb learning model discussed in Chapter 18 regularizes verbs.

Other Creative Errors

The model made a number of other interesting "errors." Its response to *The doll moved* was a particularly striking example. Recall that the training stimuli contained sentences from the frames *The animal moved*, *The human moved*, and *The object moved*. In the first two cases, the case frame contained the subject as both Agent and Patient, as in *The animal moved itself*. In the third case, the case frame contained the subject only as Patient. The model had some difficulty with these constructions, but generally put inanimate subjects into the Patient role only (as in *The desk moved*), and animate subjects into both the Agent and Instrument role. With *The doll moved*, however, the case-frame representation shows considerable activation in the Agent slot. The pattern of activation there (Figure 15) seems to be pretty much that of a small (though fragile and nonhuman) girl -- or perhaps it is simply an animate doll.

Generalization to Novel Words

We have already seen that the model can generalize quite well from particular sentences that it has learned about to new ones that it has not seen before. It is, obviously, important to be able to generalize in this way, since we cannot expect the training set of sentences to cover all possible word combinations in the language, even if they contain only a single clause. Thus far, however, we have only considered generalization to sentences made up of familiar words. What would happen, we wondered, if we tested the model using new words that it had never seen before? To examine this issue, we tested the model on the verb *touch*. *Touch* differs from *hit* in only one feature, namely, intensity; otherwise all of the same verb features are appropriate to it. We assumed that the model already knew the meaning of *touch* -- that is, we assumed that the correct input microfeatures were activated on presentation of a sentence containing it. We then took all of the test sentences containing the word *hit*, replaced *hit* with *touched*, and tested the model on each of these sentences. Overall, performance was somewhat worse than with *hit*, but the model was still able to assign arguments to the correct

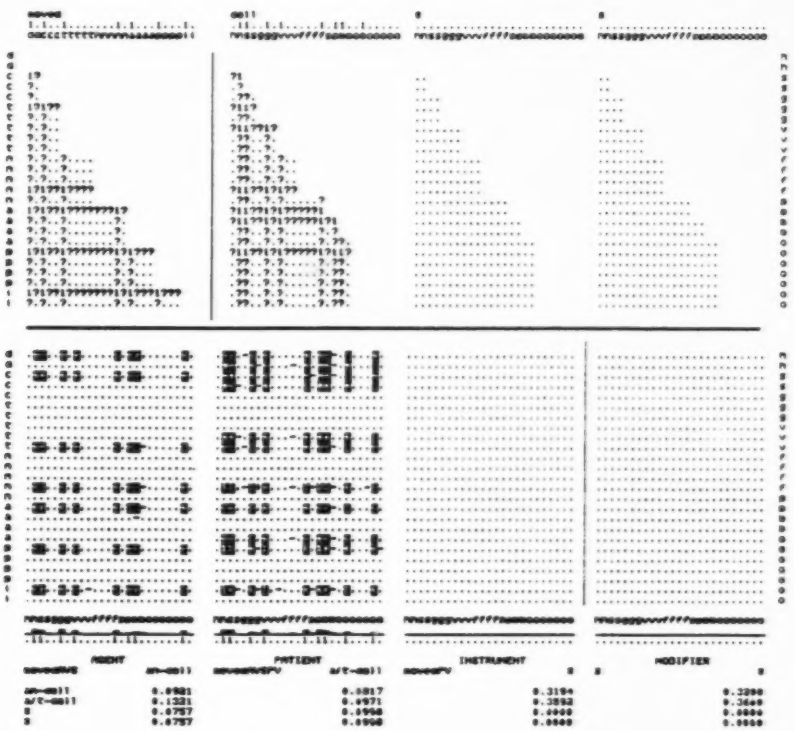

Figure 15. State of the model after processing the sentence *The doll moved*. The units with the black background in the Agent slot indicate the units appropriate for the pattern corresponding to an "animate doll" as Agent of the Agent-Verb-Self (AVS) reading of *move*. The units with the black background in the Patient slot indicate the units appropriate for either an animate or (normal) toy doll as Patient of either the AVS or Patient-Verb (PV) reading of *move*.

underlying roles. In particular, it had no difficulty assigning animate Subject NPs to the role of Agent and the Object NP to the role of Patient, nor did it have any problem assigning inanimate subjects to the role of Instrument, as illustrated in Figure 16.

Two characteristics of the model's performance with this novel verb should be noted. The first is that it does not activate all the correct verb features: The verb features captured in the case role representation are appropriate for *hit* rather than *touch*. There are two reasons for this. One is the redundancy of the representation of the verbs. The input representation we are using is an extremely rich and redundant representation for capturing the very small number of verbs that we have actually used, so the model learns to predict features from other features. In a much less redundant set of verbs, the model would be forced to learn to rely on

Figure 16. Summary of the pattern of activation produced by the model in response to *The hatchet touched the pasta* and *The girl touched the boy*.

just the right features of the sentence-level pattern to predict the features of the case role representation correctly. The second reason is that the deviant feature on the word *touch* does not fall inside a subspace delineated by a set of related exemplars that behave in similar ways. What we are seeing here, essentially, is assimilation to a single familiar verb that is near the novel verb in feature space. When a novel word is presented that more nearly falls inside the subspace delineated by a set of related exemplars, the model is better able to correctly represent all the features of the novel word in the case role representation. To show this, we defined a new fragile object, a *lamp*, that was nonhuman, hard, neuter, medium-sized, one-dimensional, not pointed, fragile, and furniture. This set of features matches at least one of the breakable objects on every dimension, but matches none of them on all of the dimensions. We then tested the model by taking all of the familiar test sentences containing any of the familiar breakable objects (*window*, *vase*, and *plate*) and then testing the model on these sentences with the word *lamp* in the place of the familiar breakable object. In many cases, as in the first example shown in Figure 17, the model activated all the features correctly. That is, the correct feature for *lamp* on each dimension was more strongly activated than any other. The model was most vulnerable to error on the VOLUME and FORM dimensions, as in the second example shown, where it differed from two of the three familiar fragile objects in each case. The example indicates, then, that the desirable tendency of the model to shade the meanings of the words it encounters to fit the typical features it expects in certain contexts need not lead to severe distortions of novel words, as long as their feature patterns fall inside the space spanned by a set of similar words that are treated by the model in a consistent way.

Straightforward Extensions

The model does many things well, but it is really just the first step toward a PDP model of sentence processing. In the general discussion we will consider some of the more general issues that confront the attempt to extend a model of this type further. Here, we will just

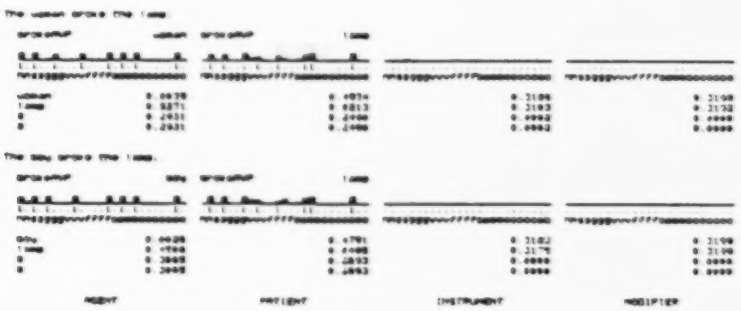

Figure 17. Summary of the response of the model to *The woman broke the lamp* and *The boy broke the lamp*.

mention some straightforward extensions that should be fairly easy to implement. We have not yet implemented these changes, so our discussion is somewhat speculative.

One straightforward set of extensions to the model would involve adding cross-connections among the case role units. The cross-connections would allow argument co-occurrence constraints that are verb-frame specific to be captured more effectively than they can now. In the present version of the model, the influence of one argument on another can only essentially add together with the influence of the verb on that argument. This will ultimately prove insufficient, we believe, because different verbs carry with them radically different interdependencies among their arguments. For example, *see* imposes some restrictions between the Instrument and the Object (as in *Bill saw the man with the telescope* and *Bill saw the man with the revolver*; the first is ambiguous, the second is not) that are different from those posed by *hit* (*Bill hit the man with the telescope* and *Bill hit the man with the revolver* seem equally ambiguous). The present version of the model would not easily capture these kinds of verb-contingent argument constraints because each argument is encoded separately from the verb at the sentence level, and it is only connections from units at the sentence level that determine the input to the case role units. This means that conjunctions of noun characteristics with verb characteristics are not directly at work in determining case role unit activations. The case role units, however, do provide just such a conjunctive encoding. It seems likely, therefore, that connections among these units would be able to capture verb-specific (or, more exactly, verb-feature specific) contingencies between slot fillers and their assignments to roles.

Cross-connections among the case role units would add a number of other advantages, as well. They would allow competition among alternative interpretations of the same word at the case-frame level, so that the stronger of two competing interpretations of the same word could effectively suppress the weaker.

A second straightforward extension would be the addition of back-connections from the case role units to the sentence-structure units. This, too, could have several beneficial effects on the performance of the model. In an extended version of the model with cross-connections and back-connections, the computation performed by the present version of the model would be just the first step in an iterative settling process. This settling process could be used to fill in the features of one reading or another of an ambiguous word at the sentence level, based on the emerging pattern at the case role level. Once filled in, these features could then add to the support for the "dominant" reading over other, initially partially activated readings -- the whole network would, in effect, drive itself into a stable "corner" that would tend to represent a coherent interpretation at both the sentence and the case role level. Kawamoto (1985) has observed just such effects in a simulation of word disambiguation based on the brain-state-in-a-box model of J. A. Anderson, Silverstein, Ritz, and Jones (1977). (Cottrell, 1985, and Waltz & Pollack, 1985, have also observed such effects in their more localist sentence-processing models.)

Back connections would also allow case role activations to actually specify the semantic features of novel or unfamiliar words occurring in constraining contexts. Consider, for example, the sentence, *The girl broke the shrafe with the feather*. The context provides a considerable amount of constraint on the properties of *shrafe*. The existing version of the model is able to fill in a plausible interpretation at the case level, but with feedback it would be able to pass this information back to the sentence level.

Another way of passing information back to the surface-structure level would be to use the back-propagation learning algorithm. The use of back propagation to train the sentence-structure units could allow the right features to be constructed at the surface level with only a phonological representation of the words as the predefined input. Back propagation might also allow us to cut down on the rather excessive numbers of units currently used in the surface-structure level. Right now there are far more SS units than are strictly necessary to do the work that the model is doing. While many units could be eliminated in a straightforward way (e.g., many of the sentence units could be eliminated because they stand for unlikely conjunctions of features across dimensions, such as *human and tool*), many more are simply redundant ways of encoding the same information and so could be consolidated into fewer units. On the other hand, for a larger vocabulary, some conjunctive units will turn out to be necessary, and pair-wise conjunctive units ultimately will probably not suffice. Indeed, we feel quite sure that no predefined coding scheme of the kind we have used could provide a sufficient basis for learning all the sentences in any real language without being immensely wasteful, so it will become crucial to train the sentence and case role units to represent just the needed conjunctions of features.

Distributed Representation of Roles

We mention a final "straightforward" extension of the model under a separate heading because it is both more speculative and perhaps somewhat more difficult to understand than the previous suggestions. This is the idea of using a distributed representation of the roles. The idea was first suggested by Hinton (1981a), and is currently under exploration by Derthick and Hinton at Carnegie-Mellon University. The essence of the idea is to think of roles not as strictly separate, monolithic objects, but as sets of role descriptors. Thus the role of Agent has certain properties: It specifies an active participant in the scenario, one that may be volitional; it specifies the instigator of the action. The role of Patient, on the other hand, specifies a passive participant, one whose volitional involvement is (pretty much) irrelevant, but who is the one that experiences the effects of the action.

Various problems arise with treating these roles as unitary objects. One is that some but not all of the Patient properties generally hold for the role nominally identified as Patient. Similarly, some but not all of the Agent properties generally hold for the role nominally identified as Agent. In certain cases, as with sentences like *The boy moved*, enough of these properties hold that we were led to assign *the boy* to both roles at once.

One suggestion that has often been made is to proliferate separate roles to deal separately with each of the slight variants of each of the traditional cases. This leads, of course, to a proliferation of roles that is ungainly, unwieldy, and inelegant, and that detracts considerably from the utility of the idea of roles as useful descriptive constructs.

Here, distributed representation can provide an elegant solution, just as it has in other instances where there appears to be a temptation to proliferate individualized, unitary representational constructs (see Chapter 17). If each role is represented by a conjunction of role properties, then far more distinct roles can be represented on the same set of role units. Furthermore, what the Agent roles of two verbs have in common is captured by the overlap of the role features in the representations of their roles, and how they differ is captured by their differences. The notion of a role that represents a combined Agent/Patient as in *The boy moved* is no longer a special case, and we get out of assigning the same argument to two different slots.

So far, the vision outstrips the implementation of this idea, but we will sketch briefly one very rudimentary instantiation of it, in what really amounts to a rather slight modification of the model we have described above. Currently, our case role units stand for conjunctions of a role, a feature of the verb, and a feature of the filler of the role. The suggestion, quite simply, is to replace these units with units that stand for a feature of a role, a feature of the verb, and a feature of the filler of the role. The first NP in *The boy broke the window with the hammer* will produce a pattern of activation over one set of these triples (corresponding pretty much to the canonical features of agenthood), while *the boy* in *The boy moved* would activate some

units from other role feature sets, as well as many of the typical agent feature units.

Again, we stress that we do not yet have much experience using this kind of distributed representations of roles. However, Derthick and Hinton (personal communication, 1985) are exploring these ideas in the context of a PDP implementation of the representation language KL-TWO (Brachman & Schmolze, 1985; Moser, 1983). They have already shown that at least one version of the idea can be made to work and that the coarse coding of roles can be used to allow inheritance of constraints on role fillers.

DISCUSSION

Now that we have examined the model in some detail and considered some possible extensions of it, we turn to more general considerations. We consider three issues. First, we examine the basic principles of operation of the model and mention briefly why they are important and useful principles for a sentence-processing model to embody. Second, we consider some of the implications of the model for thinking about language and the representation of language. Third, we address the limitations of the present model. This part of the discussion focuses on a key question concerning the feasibility of our approach, namely, the requirement that any plausible model of language processing must be able to handle sentences containing embedded clauses.

Basic Features of the Model

We emphasize before we begin that the basic features of the present model are shared with a number of other distributed models, especially those of Hinton (1981a) and those described in Chapters 3, 14, 17, and 18 of this book. The two most important properties of the model are its ability to exploit the constraints imposed by all the arguments in a sentence simultaneously and its ability to represent *shades* of meaning. These aspects are basic, we believe, to any attempt to capture the flexibility and context-sensitivity of comprehension.

The first of these properties is, perhaps, just as easily capturable using local rather than distributed connectionist networks. These local connectionist models capture this property much more effectively than they have been captured in nonconnectionist mechanisms (e.g., Small's, 1980, word expert parser; cf. Cottrell & Small, 1983). Such networks have been applied to sentence processing, particularly to the problems of ambiguity resolution and role assignment (Cottrell, 1985; Waltz & Pollack, 1985). Both models use single units to stand for alternative meanings of words or as "binders" to tie words to alternative roles, and use mutual activation and inhibition to select between alternative meanings and alternative role assignments.

The present model exhibits many of these same properties, but uses distributed representations rather than local ones. What the distributed representations have that local representations lack is the natural ability to represent a huge palette of shades of meaning. With

distributed representations, it is quite natural to represent a blend of familiar concepts or a shaded version of a familiar concept that fits a scenario (Waltz & Pollack, 1985, do this with their context microfeatures). Perhaps this is the paramount reason why the distributed approach appeals to us. To be sure, it is possible to represent different shades of meaning in a localist network. One can, for example, have different units for each significantly different variation of the meaning of a word. A problem arises, though, in specifying the meaning of "significantly different." We will probably all agree that there are different readings of the word *bat* in the sentences *The bat hit the ball* and *The bat flew round the cave*. But what about the word *chicken* in the sentences *The woman ate the chicken* and *The wolf ate the chicken*? Or what about the word *ball* in *The baby kicked the ball* and *The ball broke the window*? There is no doubt that we think of different balls in these cases; but do we really want to have a separate unit in memory for the soft, squishy, rubber ball the baby kicks and the small, hard ball that can break a window?

With distributed representations, we do not have to choose. Different readings of the same word are just different patterns of activation; really different readings, ones that are totally unrelated, such as the two readings of *bat* simply have very little in common. Readings that are nearly identical with just a shading of a difference are simply represented by nearly identical patterns of activation.

These properties of distributed representations are extremely general, of course, and they have come up before, particularly in the chapter on schemata (Chapter 14). We also just invoked them in suggesting that we might be able to use distributed representations instead of some fixed set of case roles. In both of these other cases, as in the case of distributed representation of word senses, the use of distributed representation allows for all shades and degrees of similarity and difference in two representations to be captured in a totally seamless way.

A final basic feature of the model is the gradualness of acquisition it exhibits. We have not stressed the time course of acquisition, but it was, of course, a crucial property of the verb learning model, described in Chapter 18, and it is quite evident that acquisition is gradual from Figure 4. As with the verb learning model, our model also seems to pick up on the strongest regularities first. This is seen most easily in Figures 5 and 6 by comparing NVN sentences from the *hit* and *broke* generators. Those with animate preverbal NPs, which are Agents, are learned more quickly than those with inanimate preverbal NPs, which are Instruments. This is because a far greater number of constructions have animate, Agent subjects than have inanimate, Instrument subjects.

These three basic properties -- exploitation of multiple, simultaneous constraints, the ability to represent continuous gradations in meaning, and the ability to learn gradually, without formulating explicit rules, picking up first on the major regularities, are hallmarks of parallel distributed models, and they are no less applicable to comprehension of language than

they are to any other aspect of cognitive processes.

Do Words Have Literal Meanings?

There is one further aspect of the distributed approach to representation of meaning that should be mentioned briefly. This is the stand our model takes on the issue of whether words have literal meanings. It is normal and natural to think of words as having literal meanings, but it is very difficult to say what these meanings really are. For, as we have noted throughout this chapter, the apparent meanings of words are infinitely malleable and very difficult to pin down. An alternative view is that words are clues to scenarios. This view, which has been proposed by Rumelhart (1979) among others, never made very much impression on us until we began to study the present model. However, in exploring the model, we have found that it embodies Rumelhart's idea exactly. A sentence assembles some words in a particular order, and each provides a set of clues that constrains the characteristics of the scenario, each in its own way. The verb, in and of itself, may specify a range of related scenarios and certain constraints on the players. The nouns further restrict the scenario and further constrain the players. But the words themselves are no longer present in the scenario, nor is there necessarily anything in the scenario that corresponds to the literal meaning of any of the words. Thus in the case of *The doll moved*, the (partially activated) Agent is not a copy of the standard doll pattern, but a pattern appropriate for a doll that can move under its own steam.

The crucial point, here, is that *all* the words work together to provide clues to the case frame representation of the sentence, and *none* of the words uniquely or completely determine the representation that is assigned to any of the constituents of the underlying scenario. Certainly, the word *hammer* most strongly constrains the filler of the Instrument role in *The boy broke the vase with the hammer*, but the other words contribute to the specification of the filler of this role, and *hammer* contributes to the specification of the fillers of the other roles. Compare *The prisoner struck the rock with the hammer* and *The boy broke the vase with the feather*: The former suggests a heavier hammer; the latter suggests an extremely fragile vase (if we give an instrumental reading to the with-NP).

Toward a More Complete Model of Sentence Processing

As we have already made clear, the model that we have described in this chapter is far from a complete model of the psychological processes involved in sentence processing. It does not deal with the fact that sentence processing is an on-line process, a process that unfolds in real time as each word is heard. It does not deal with the integration of processed sentences into larger contextual frameworks. It does not handle anaphora and other referential phenomena, or tense, aspect, or number. No attempt is made to deal with quantification or scoping issues. The model even lacks a way of distinguishing different tokens with identical featural descriptions. Thus it does not explicitly designate separate dogs in *dog eat dog* and only one dog in *The dog chased himself*. Finally, the model completely ignores the

complexities of syntax. For the present model, a sentence can come in only one rigidly structured form, and no embedded clauses, cleft sentences, or passive constructions are allowed.

Clearly, we have much work to do before we can claim to have a model that is in any sense complete or adequate. The question is, can it be done at all? Is there any fundamental limitation in the PDP approach that will prevent the successful development of a full-scale model of language processing that preserves the positive aspects of the distributed approach?

Obviously, the proof is in the pudding. But we think the enterprise can succeed. Rather than discuss all of the issues raised above, we will discuss one that seems quite central, namely, the application of PDP models to the processing of sentences with embedded clauses. We consider several different ways that PDP models could be applied to the processing of such sentences.

Interfacing PDP mechanisms with conventional parsers. We start with what might be the simplest view, or at any rate the most conventional: the idea that a model such as ours might be interfaced with a conventional parser. For example, we might imagine that a parser similar to Marcus's PARSIFAL (1980) might pass off the arguments of completed (or possibly even incomplete) clauses to a mechanism such as the one we have proposed for case role assignment and PP attachment. In this way, the "role-assignment module" could be used with any given sentoid and could be called repeatedly during the processing of a sentence containing embedded clauses.

Interfacing our model with a conventional parser would perhaps provide a way of combining the best of both conventional symbol processing and parallel distributed processing. We are not, however, particularly inclined to follow this route ourselves. For it appears that it will be possible to implement the parser itself as a PDP mechanism. As we shall see, there are at least three ways this might be done. One involves implementing a true recursive automaton in a PDP network. We describe this method first, even though we suspect that the human parser is not in fact such a machine. After describing the mechanism, we will explain our objections to this view of the human sentence-processing mechanism. This will lead us to suggest two other mechanisms. One relies on the connection information distribution mechanism described in Chapter 16 to program a parallel net to process sentences of indefinite length and embeddedness; the other operates iteratively rather than recursively. It is more computationally limited in some respects than the other mechanisms, but the limitations appear to conform to those of the human parser, as we shall see.

A PDP model that does recursion. It turns out that it is not difficult to construct a parallel network that does true recursive processing. Hinton (personal communication) worked out the scheme we will describe here and implemented a rudimentary version of it in 1973. While such a mechanism has not been applied directly to parsing, the fact that recursive processing is possible suggests that there is no reason, in principle, why it should not provide a

sufficient basis for implementing some kind of parser.

The mechanism consists of a large network of units. Patterns of activation on these units are distributed representations of a particular state in a processing sequence. Processing occurs through the succession of states. The units are divided up into subnets that represent particular parts of the state. One important subnet is a set of units that provides a distributed pattern corresponding to a stack-level counter.

The connections in the network are set up so that successive states of a routine are associated with their predecessors. Thus, when one state is in place, it causes another to follow it. States may also be used to drive actions, such as output of a line segment, say, if the automaton is a mechanism for recursively drawing figures, as Hinton's was. Processing, then, amounts to carrying out a sequence of states, emitting actions (and possibly reading input) along the way.

Calling subroutines in such a mechanism is not particularly difficult, since all that is required is to associate a particular state (the calling state) with the start state of some routine. Passing parameters to the called routine is likewise not particularly difficult; in the simplest case they can be parts of the calling state that are carried along when the routine is called.

To implement recursion in such a network, all that is required is a way to reinstate the calling state when a routine is done. To do this, the mechanism associates the state of the stack-level units with the state that is in place over the rest of the units, using an associative learning rule to adjust connection strengths while processing is taking place. These associations are implemented by rapidly changing a short-term component of the weights whose long-term values implement the associations that allow the model to cycle through a sequence of states. The temporary associations stored in these short-term weights are not strong enough to overrule the long-term weights, but they are sufficiently strong to determine the next state of the network when the long-term weights leave several possibilities. So, at the end of a routine at stack level N, the network associatively reinstates stack level $N - 1$, with the rest of the state cleared. This associative reinstatement of the previous stack-level state would be based on long-term, relatively permanent associations between states corresponding to adjacent depths in the stack. This reinstated stack-level state, which was associated in the short-term weights with the calling state just before the subroutine call occurred, would then simply use this short-term association to reinstate the pattern that existed before the call. (A "done" bit would have to be set to keep the process from doing the call again at this point.)

There is no apparent a priori limit to the number of embedded calls that could be carried out by such a network, though for any fixed size of the stack-level subnet, there will be a corresponding maximum number of associations that can be learned without error. Of course, similar limitations also occur with all other stack machines; the stack is always of finite depth.

There would also likely be interference of previous calling states when returning from any particular level, unless the learning were carefully tuned so that earlier associations with a particular stack level were almost completely wiped out or decayed by the time a new one must be used. Care would also be necessary to avoid crosstalk between stack-level representations. However, these problems can be overcome by using enough units so that very different states are used to represent each level of the stack.

Drawbacks of true recursion. The scheme described in the previous section has several fairly nice properties and deserves considerably more exploration than it or some obvious variants have received to date. However, it does have one drawback from our point of view -- one that it shares with other, more conventional implementations of truly recursive automata. The drawback is that the calling state is not present and active during the subroutine call; it is effectively inaccessible until it is reinstated after the return.

This property of truly recursive schemes limits their ability to simultaneously consider binding a prepositional phrase at each of two levels of embedding. Consider the sentences:

(14) The boy put the cake the woman made in the kitchen.
(15) The boy saw the cake the woman made in the kitchen.

Our preferred reading of the first of these two sentences has the boy putting the cake in the kitchen, rather than the woman preparing it there; while in the second case, the preferred interpretation appears to be that the woman made the cake in the kitchen, and the boy saw it at some unspecified location. Since the material is the same from the beginning of the embedding in both cases, it appears that the demand the matrix clause material (*The boy put the cake . . .*) makes for a locative argument influences the decision about whether *in the kitchen* should be bound into the subordinate clause. While it may be possible to arrive at these two different readings in a conventional parser by backtracking or by passing aspects of the calling state along when the subroutine is called, it would seem to be more natural to suppose that the matrix clause is actively seeking its missing locative argument as the embedded material is being processed, and so is prepared to steal *in the kitchen* from the verb in the embedded clause. Thus, it appears that a mechanism capable of processing at different levels of embedding at the same time is needed.

A fixed-length sentence processor. A connectionist parser that can, in principle, handle this kind of competition among alternative attachment decisions at different levels of embedding has recently been developed and implemented by Fanty (1985). He describes a mechanism that effectively parses a sentence at many levels at the same time. The parser consists of a fixed network of units. Some of the units represent the terminal and nonterminal symbols of the grammar; other units, called *match units*, represent the different possible expansions of each nonterminal. The symbol units are easily represented in a table in which the columns represent starting positions in the input string and the rows represent lengths. There is a unit

for each terminal symbol in each position of the bottom row, and there is a unit for each nonterminal symbol at every position in the table; that is, there is a copy of each nonterminal unit for every possible portion of the input that it could cover. For each of these there is a set of *binder units*, one for each possible expansion of each nonterminal unit.

A simple version of the table, for the indicated three-rule grammar, is shown in Figure 18. Only a subset of the units -- the ones that would become active in the parse of *aabbb* -- are shown.

The parser can only process strings up to a predefined maximal length. Essentially, it processes the entire sentence in one two-pass processing sweep. In the first, bottom-up pass, all possible constituents are identified, and in the second, top-down pass, the constituents that fit together with the top S and the subconstituents of the top S are reinforced. These active units represent the parse tree.

A very nice feature of Fanty's parse is that it takes into account all levels of the parse tree simultaneously. This allows it to find a globally satisfactory parse in one pair of sweeps, eliminating possible constituents that do not fit together with others to make a globally acceptable structure. With some modifications (different degrees of strength for different rules; continuous, interactive processing as opposed to a single pair of sweeps), it would probably be possible to implement a mechanism that could choose the "better" of two alternative acceptable parses, as people seem to do with many ambiguous sentences. The parser of Selman and

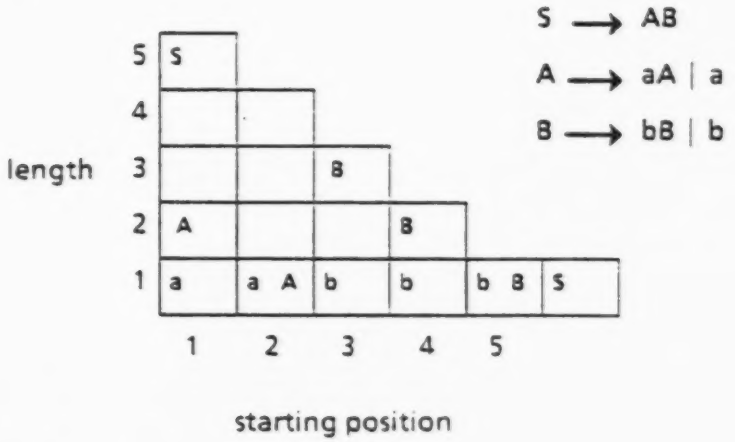

Figure 18. Parsing table used in Fanty's multilevel parser. Only the units active in the correct parse of the string *aabbb* are illustrated. (From *Context-Free Parsing in Connectionist Networks* [TR-174, p. 3] by M. Fanty, 1985, Rochester, NY: University of Rochester, Department of Computer Science. Copyright 1985 by M. Fanty. Reprinted by permission.)

Hirst (1985; Selman, 1985), which is similar to the Fanty parser in structure but uses simulated annealing, appears to have just the right characteristics in this regard.

However, this kind of parser does have some drawbacks. Most importantly, it is limited to sentences of a prespecified length. To expand it, one needs not only to add more units, one needs to program these units with the connections that allow them to do the jobs they are needed or potentially needed to do. Second, the size does grow rather quickly with the allowable length (see Fanty, 1985, for details).

Fanty's model is, in fact, somewhat reminiscent of the TRACE model of speech perception (Chapter 15) in its reduplication of dedicated units and connections. As with TRACE, it may be possible to use connection information distribution (Chapter 16) to program the necessary connections in the course of processing from a single, central network containing the system's long-term knowledge of the rules of English. Indeed, Fanty (1985) has explored an off-line variant of the connection information distribution scheme; his version learns new productions locally, and sends the results to a central network which then distributes the results to the rest of the net, off-line. If the programming of the connections could be made to work on-line in the course of processing, as with the programmable blackboard model of reading, we would have a mechanism that still needs large numbers of units and connections, but crucially, these would be *uncommitted* units and *programmable* connections. The computational capability of the machine would be expandable, simply by increasing the size of the programmable network.

Myths about the limitations of PDP models. We hope that these last two sections will help to dispel some widely held beliefs about computation and PDP models. We will briefly consider two variants of these beliefs. The first, quite simply, is that PDP models cannot do recursion. Hinton's recursive processor needs considerably more development before we will have a working simulation that proves that this belief is wrong, but it seems clear enough that a recursive PDP processor will be available fairly soon. In fact, it is likely that a slightly different approach will be explored more fully first: Touretzky and Hinton (1985) have recently developed a PDP implementation of a production system that can do rudimentary variable binding, and at present it appears that they may be able to extend it to perform recursive computations.

The second belief seems to follow from the first: It is that PDP models are inherently incapable of processing a full range of sentences. We say it only *seems* to follow because it depends upon accepting the assumption that in order to process sentences it is necessary to be able to do recursion. Most people who have thought computationally about sentence processing are familiar with sentence-processing mechanisms that are in fact recursive, such as the ATN (Woods, 1970) or the Marcus parser. Since sentences are recursively defined structures, the (implicit) argument goes, the mechanisms for parsing them must themselves operate recursively.

Fanty's (1985) parser, or an extension of it incorporating connection information distribution, begins to suggest that this may not be so. In a programmable version of Fanty's parser, we would have captured the essential property of a recursive automaton -- that the same procedural knowledge be applicable at any point in a parse tree. But we would have captured it in a very exciting new way, a way that would free the mechanism from the serial processing constraint that prevents conventional recursive mechanisms from being able to exploit constraints from many levels at the same time. Connection information distribution may actually permit us to reap the benefits of simultaneous mutual constraints while at the same time enjoying the benefits of being able to apply the same bit of knowledge at many points in processing the same sentence.

There are a couple of caveats, however. One is that connection information distribution is very expensive computationally; a considerable number of units and connections are required to handle the input to and output from the central knowledge structures (see Chapter 12 for further discussion), and the resource demands made by even a fixed version of Fanty's parser are quite high already. There may turn out to be ways of reducing the resource demands made by the Fanty parser. In the meantime, it is worth asking whether some other approach might not succeed nearly as well with fewer resources.

Context-sensitive coding, iteration, and center embedding. There is one more belief about sentences, this one even more deeply ingrained than the last, that we have tacitly accepted up to this point. This is the belief that sentences are indeed recursively defined structures. Clearly, sentences are recursively definable, but there is one proviso: Only one level of center embedding is allowed. It may be controversial whether the "true" competence grammar of English accepts multiple center-embedded sentences, but people cannot parse such sentences without the use of very special strategies, and do not even judge them to be acceptable (G. A. Miller, 1962). Consider, for example, the "sentence":

(16) The man who the girl who the dog chased liked laughed.

The unparsability of sentences such as (16) has usually been explained by an appeal to adjunct assumptions about performance limitations (e.g., working-memory limitations), but it may be, instead, that they are unparsable because the parser, by the general nature of its design, is simply incapable of processing such sentences.[13]

It should be noted that parsers like the ATN and the Marcus parser are most certainly *not* intrinsically incapable of processing such sentences. Indeed, such parsers are especially well suited for handling an indefinite number of center embeddings. Such mechanisms are

13. This argument has been made previously by a number of other authors, including Reich (1969). In fact, Reich proposed an iterative approach to sentence processing that is similar, in some respects, to the one we consider here.

clearly necessary for processing such things as Lisp expressions, such as this one taken from a program (actually, itself a parser) written by Jeff Sokolov:

```
((and (eq (car (explode arg)) '/ )
      (eq (car (reverse (explode arg))) '/ ))
 (implode (reverse (cdr (reverse (cdr (explode arg)))))))
```

But sentences in natural language are simply not structured in this way. Perhaps, then, the search for a model of natural language processing has gone down the garden path, chasing a recursive white rabbit.

One approach to sentence processing holds that we should ask much less of a syntactic parser and leave most of the work of sentence processing to a more conceptual level of processing. This position is most strongly associated with Schank (1973), and Charniak (1983) is among the recent proponents of this view.

Let us assume that the job of the parser is to spit out phrases encoded in a form that captures their local context, in a way that is similar to the way the verb learning model, described in Chapter 18, in a form that captures their local context.[14] Such a representation may prove sufficient to allow us to reconstruct the correct bindings of noun phrases to verbs and prepositional phrases to nearby nouns and verbs. In fact, we suspect that this kind of local context-sensitive encoding can capture the attachments of NPs to the right verbs in "tail-recursive" sentences like

(17) This is the farmer that kept the cock that waked the priest that married the man that milked the cow that tossed the dog that worried the cat that killed the rat that ate the malt that lay in the house that Jack built.[15]

Locally context-sensitive coding will begin to break down, however, when there is center embedding -- specifically, more than one level of center embedding. Local context can be used to signal both the beginning and the end of an embedding, but cannot be used to specify which beginning of an embedding the material just after the end of an embedding should be bound to. Thus if we read to the last word of

---

14. For this to work it will be necessary to code units, not in terms of the adjacent words, but in terms of neighboring constituents more abstractly defined. Thus, in *The girl in the hat saw the mouse on the floor*, we will want to encode the complex NP *the girl in the hat* as adjacent to the verb *saw*. Thus, local context will have to be defined, as it is in Marcus (1980), in terms of constituents, not merely in terms of words. Getting this to work will be one of the major challenges facing this approach.

15. Adapted from E. Johnson, E. R. Sickels, & F. C. Sayers (Eds.), *Anthology of Children's Literature* (4th ed., p. 16), 1970, Boston: Houghton-Mifflin.

(18) The man who the girl who the dog chased laughed

we may not know whether to bind *laughed* (just after the end of the embedding) to *the man* (NP before an embedding) or to *the girl* (a different NP before an embedding). If we bind it to *the man*, we may experience a false sense of closure -- this sentence is ungrammatical because it has only two verbs for three clauses.

These suggestions lead us to the following conjecture: It may be possible to build a PDP language-processing mechanism that works iteratively along a sentence, building constituents represented by distributed patterns conjunctively encoded with their local context of occurrence. The mechanism would need a way of unifying constituents on two sides of an intervening embedding. Exactly how this would be done remains to be established, but as long as it is done in terms of a mechanism sensitive only to the local context of the constituents before and after the embedding, it may succeed where there is a single embedding but fail in multiply embedded sentences where there are two suspended, incomplete constituents that the mechanism must choose between completing.

We hope it is clear that these ideas are speculative and that they are but pointers to directions for further research. Indeed, all three of the directions we have described in this section are only just beginning to be explored systematically, and it is unclear which of them will prove most attractive on closer scrutiny. We mention them because they suggest that ways of overcoming some of the apparent limitations of PDP mechanisms may not be very far beyond our present grasp, and that it may soon be possible to retain the benefits of parallel distributed processing in mechanisms that can cope with the structural complexity and semantic nuance of natural language.

References

Anderson, J. A. (1983). Cognitive and psychological computation with neural models. *IEEE Transactions on Systems, Man, and Cybernetics, SMC-13*, 799-815.

Anderson, J. A., Silverstein, J. W., Ritz, S. A., & Jones, R. S. (1977). Distinctive features, categorical perception, and probability learning: Some applications of a neural model. *Psychological Review, 84*, 413-451.

Bates, E., & MacWhinney, B. (in press). Competition, variation, and language learning: What is not universal in language acquisition. In B. MacWhinney (Ed.), *Mechanisms of language acquisition*. Hillsdale, NJ: Erlbaum.

Bever, T. G. (1970). The cognitive basis for linguistic structures. In J. R. Hayes (Ed.), *Cognition and the development of language*. New York: Wiley.

Brachman, R. J., & Schmolze, J. G. (1985). An overview of the KL-ONE knowledge representation system. *Cognitive Science, 9*, 171-216.

Charniak, E. (1983). Passing markers: A theory of contextual influence in language comprehension. *Cognitive Science, 7*, 171-190.

Cottrell, G. (1985). *A connectionist approach to word sense disambiguation* (TR-154). Rochester, NY: University of Rochester, Department of Computer Science.

Cottrell, G., & Small, S. (1983). A connectionist scheme for modeling word sense disambiguation. *Cognition and Brain Theory, 1*, 89-120.

Crain, S., & Steedman, M. (1985). On not being led up the garden-path: The use of context by the psychological parser. In D. Dowty, L. Karttunen, & A. Zwicky (Eds.), *Natural language parsing*. Cambridge, England: Cambridge University Press.

Fanty, M. (1985). *Context-free parsing in connectionist networks* (TR-174). Rochester, NY: University of Rochester, Department of Computer Science.

Fillmore, C. J. (1968). The case for case. In E. Bach & R. T. Harms (Eds.), *Universals in linguistic theory*. New York: Holt, Rinehart, & Winston.

Fodor, J. A., Bever, T. G., & Garrett, M. F. (1974). *The psychology of language*. New York: McGraw-Hill.

Ford, M., Bresnan, J., & Kaplan, R. M. (1982). A competence-based theory of syntactic closure. In J. Bresnan (Ed.), *The mental representation of grammatical relations.* Cambridge, MA: MIT Press.

Frazier, L., & Rayner, K. (1982). Making and correcting errors during sentence comprehension: Eye movements in the analysis of structurally ambiguous sentences. *Cognitive Psychology, 14,* 178-210.

Hinton, G. E. (1981f). Implementing semantic networks in parallel hardware. In G. E. Hinton & J. A. Anderson (Eds.), *Parallel models of associative memory* (pp. 161-188). Hillsdale, NJ: Erlbaum.

Kaplan, R., & Bresnan, J. (1982). Lexical functional grammar: A formal system for grammatical representation. In J. Bresnan (Ed.), *The mental representation of grammatical relations.* Cambridge, MA: MIT Press.

Kawamoto, A. H. (1985). *Dynamic processes in the (re)solution of lexical ambiguity.* Unpublished doctoral dissertation, Brown University.

Kurtzman, H. S. (1985). *Studies in syntactic ambiguity resolution.* Unpublished doctoral dissertation, Massachusetts Institute of Technology.

MacWhinney, B., Bates, E., & Kliegl, R. (1984). Cue validity and sentence interpretation in English, German, and Italian. *Journal of Verbal Learning and Verbal Behavior, 23,* 127-150.

MacWhinney, B., & Sokolov, J. L. (in press). The competition model of the acquisition of syntax. In B. MacWhinney (Ed.), *Mechanisms of language acquisition.* Hillsdale, NJ: Erlbaum.

Marcus, M. P. (1980). *A theory of syntactic recognition for natural language.* Cambridge, MA: MIT Press.

Marslen-Wilson, W. D., & Tyler, L. K. (1981). Central processes in speech understanding. *Philosophical Transactions of the Royal Society, London B, 295,* 317-332.

Miller, G. A. (1962). Some psychological studies of grammar. *American Psychologist, 17,* 748-762.

Moser, M. G. (1983). An overview of NIKL: The new implementation of KL-ONE. In C. Sidner, M. Bates, R. Bobrow, B. Goodman, A. Haas, R. Ingria, D. Israel, D. McAllester, M. Moser, J. Schmolze, & M. Vilain (Eds.), *Research in knowledge representation for*

natural language understanding: Annual report (BBN Report No. 5421). Cambridge, MA: Bolt, Beranek, and Newman.

Oden, G. C. (1978). Semantic constraints and judged preference for interpretations of ambiguous sentences. *Memory & Cognition, 6*, 26-37.

Rayner, K., Carlson, M., & Frazier, L. (1983). The interaction of syntax and semantics during sentence processing: Eye movements in the analysis of semantically biased sentences. *Journal of Verbal Learning and Verbal Behavior, 22*, 358-374.

Reich, P. A. (1969). The finiteness of natural language. *Language, 45*, 831-843.

Rosenblatt, F. (1962). *Principles of neurodynamics.* New York: Spartan.

Rumelhart, D. E. (1979). Some problems with the notion of literal meanings. In A. Ortony (Ed.), *Metaphor and thought.* Cambridge, England: Cambridge University Press.

Schank, R. C. (1973). Identification of conceptualizations underlying natural language. In R. C. Schank & K. M. Colby (Eds.), *Computer models of thought and language.* San Francisco: Freeman.

Selman, B. (1985). *Rule-based processing in a connectionist system for natural language understanding* (TR CSRI-168). Toronto: University of Toronto, Computer Systems Research Institute.

Selman, B., & Hirst, G. (1985). A rule-based connectionist parsing system. *Proceedings of the Seventh Annual Conference of the Cognitive Science Society*, 212-221.

Small, S. L. (1980). *Word expert parsing: A theory of distributed word-based natural language understanding* (TR- 954). Baltimore: University of Maryland, Department of Computer Science.

Touretzky, D., & Hinton, G. E. (1985). Symbols among the neurons: Details of a connectionist inference architecture. *Proceedings of the Ninth International Joint Conference on Artificial Intelligence.*

Waltz, D. L., & Pollack, J. B. (1985). Massively parallel parsing. *Cognitive Science, 9*, 51-74.

Woods, W. (1970). Transition network grammars for natural language analysis. *Communications of the ACM, 13*, 591-606.

Dr. Phillip L. Ackerman
University of Minnesota
Department of Psychology
Minneapolis, MN 55455

Air Force Human Resources Lab
AFHRL/MPD
Brooks AFB, TX 78235

AFOSR,
Life Sciences Directorate
Bolling Air Force Base
Washington, DC 20332

Dr. Robert Ahlers
Code N711
Human Factors Laboratory
NAVTRAEQUIPCEN
Orlando, FL 32813

Dr. Ed Aiken
Navy Personnel R&D Center
San Diego, CA 92152

Dr. Earl A. Alluisi
HQ, AFHRL (AFSC)
Brooks AFB, TX 78235

Dr. James Anderson
Brown University
Center for Neural Science
Providence, RI 02912

Dr. John R. Anderson
Department of Psychology
Carnegie-Mellon University
Pittsburgh, PA 15213

Dr. Nancy S. Anderson
Department of Psychology
University of Maryland
College Park, MD 20742

Technical Director, ARI
5001 Eisenhover Avenue
Alexandria, VA 22333

Dr. Alan Baddeley
Medical Research Council
Applied Psychology Unit
15 Chaucer Road
Cambridge CB2 2EF
ENGLAND

Dr. Jackson Beatty
Department of Psychology
University of California
Los Angeles, CA 90024

Dr. Alvah Bittner
Naval Biodynamics Laboratory
New Orleans, LA 70189

Dr. John Black
Yale University
Box 11A, Yale Station
New Haven, CT 06520

Dr. Arthur S. Blaives
Code N711
Naval Training Equipment Center
Orlando, FL 32813

Dr. Gordon H. Bower
Department of Psychology
Stanford University
Stanford, CA 94306

Dr. Robert Breaux
Code N-095R
NAVTRAEQUIPCEN
Orlando, FL 32813

Dr. Gail Carpenter
Northeastern University
Department of Mathematics, 504LA
360 Huntington Avenue
Boston, MA 02115

Dr. Pat Carpenter
Carnegie-Mellon University
Department of Psychology
Pittsburgh, PA 15213

Chair, Department of
Psychology
College of Arts and Sciences
Catholic University of
America
Washington, DC 20064

Dr. Fred Chang
Navy Personnel R&D Center
Code 51
San Diego, CA 92152

Dr. David E. Clement
Department of Psychology
University of South Calorina
Columbia, SC 29208

Dr. Charles Clifton
Tobin Hall
Department of Psychology
University of
Massachusetts
Amherst, MA 01003

Dr. Michael Coles
University of Illinois
Department of Psychology
Champaign, IL 61820

Dr. John J. Collins
Director, Field Research
Office, Orlando
NPRDC Liaison Officer
NTSC Orlando, FL 32813

Dr. Stanley Collyer
Office of Naval Technology
Code 222
800 N. Quincy Street
Arlington, VA 22217-5000

Dr. Leon Cooper
Brown University
Center for Neural Science
Providence, RI 02912

CAPT P. Michael Curran
Office of Naval Research
800 N. Quincy St.
Code 125
Arlington, VA 22217-5000

Bryan Dallman
AFHRL/LRT
Lowry AFB, CO 80230

Dr. Joel Davis
Office of Naval Research
Code 1141NP
800 North Quincy Street
Arlington, VA 22217-5000

Dr. R. K. Dismukes
Associate Director for Life
Science
AFOSR
Bolling AFB
Washington, DC 20332

Dr. Emanuel Donchin
University of Illinois
Department of Psychology
Champaign, IL 61820

Defense Technical
Information Center
Cameron Station, Bldg 5
Alexandria, VA 22314
Attn: TC
(12 Copies)

Dr. Ford Ebner
Brown University
Anatomy Department
Medical School
Providence, RI 02912

Dr. Jeffrey Elman
University of California,
San Diego
Department of Linguistics, C-008
La Jolla, CA 92093

Dr. William Epstein
University of Visconsin
W. J. Brogden Psychology Bldg.
1202 W. Johnson Street
Madison, WI 53706

ERIC Facility-Acquisitions
4833 Rugby Avenue
Bethesda, MD 20014

Dr. K. Anders Ericsson
University of Colorado
Department of Psychology
Boulder, CO 80309

Dr. W. E. Evans
Hubbs Sea World Institute
1720 S. Shores Rd.
Mission Bay
San Diego, CA 92109

Dr. Martha Farah
Department of Psychology
Carnegie-Mellon University
Schenley Park
Pittsburgh, PA 15213

Dr. Jerome A. Feldman
University of Rochester
Computer Science Department
Rochester, NY 14627

J. D. Fletcher
9931 Corsica Street
Vienna VA 22180

Dr. John R. Frederiksen
Bolt Beranek & Newman
50 Moulton Street
Cambridge, MA 02138

Dr. Michaela Gallagher
University of North Carolina
Department of Psychology
Chapel Hill, NC 27514

Dr. Michael Genesereth
Stanford University
Computer Science Department
Stanford, CA 94305

Dr. Don Gentner
Center for Human
Information Processing
University of California
La Jolla, CA 92093

Dr. Claude Ghez
Center for Neurobiology and
Behavior
722 W. 168th Street
New York, NY 10032

Dr. Gene L. Gloye
Office of Naval Research
Detachment
1030 E. Green Street
Pasadena, CA 91106-2485

Dr. Sam Glucksberg
Princeton University
Department of Psychology
Green Hall
Princeton, NJ 08540

Dr. Daniel Gopher
Industrial Engineering
& Management
TECHNION
Haifa 32000
ISRAEL

Dr. Sherrie Gott
AFHRL/MODJ
Brooks AFB, TX 78235

Jordan Grafman, Ph.D.
Department of Clinical
Investigation
Walter Reed Army Medical Center
6825 Georgia Ave., N. W.
Washington, DC 20307-5001

Dr. Richard H. Granger
Department of Computer Science
University of California, Irvine
Irvine, CA 92717

Dr. Wayne Gray
Army Research Institute
5001 Eisenhover Avenue
Alexandria, VA 22333

Dr. William Greenough
University of Illinois
Department of Psychology
Champaign, IL 61820

Dr. Stephen Grossberg
Center for Adaptive Systems
Room 244
111 Cummington Street
Boston University
Boston, MA 02215

Dr. Muhammad K. Habib
University of North Carolina
Department of Biostatistics
Chapel Hill, NC 27514

Dr. Henry M. Halff
Halff Resources, Inc.
4918 33rd Road, North
Arlington, VA 22207

Dr. Ray Hannapel
Scientific and Engineering
Personnel and Education
National Science Foundation
Washington, DC 20550

Stevan Harnad
Editor, The Behavioral and
Brain Sciences
20 Nassau Street, Suite 240
Princeton, NJ 08540

Dr. Steven A. Hillyard
Department of Neurosciences
University of California,
San Diego
La Jolla, CA 92093

Dr. Geoffrey Hinton
Carnegie-Mellon University
Computer Science Department
Pittsburgh, PA 15213

Dr. Jim Hollan
Intelligent Systems Group
Institute for
Cognitive Science (C-015)
UCSD
La Jolla, CA 92093

Dr. John Holland
University of Michigan
2313 East Engineering
Ann Arbor, MI 48109

Dr. Keith Holyoak
University of Michigan
Human Performance Center
330 Packard Road
Ann Arbor, MI 48109

Dr. Earl Hunt
Department of Psychology
University of Washington
Seattle, WA 98105

Dr. Alice Isen
Department of Psychology
University of Maryland
Catonsville, MD 21228

Chair, Department of
Psychology
The Johns Hopkins University
Baltimore, MD 21218

Dr. Marcel Just
Carnegie-Mellon University
Department of Psychology
Schenley Park
Pittsburgh, PA 15213

Dr. Daniel Kahneman
The University of British Columbia
Department of Psychology
#154-2053 Main Mall
Vancouver, British Columbia
CANADA V6T 1Y7

Dr. Demetrios Karis
Grumman Aerospace Corporation
MS C04-14
Bethpage, NY 11714

Dr. Milton S. Katz
Army Research Institute
5001 Eisenhover Avenue
Alexandria, VA 22333

Dr. Steven W. Keele
Department of Psychology
University of Oregon
Eugene, OR 97403

Dr. Wendy Kellogg
IBM T. J. Watson Research Ctr.
P.O. Box 218
Yorktown Heights, NY 10598

Dr. Scott Kelso
Haskins Laboratories,
270 Crown Street
New Haven, CT 06510

Dr. David Klahr
Carnegie-Mellon University
Department of Psychology
Schenley Park
Pittsburgh, PA 15213

Dr. Sylvan Kornblum
University of Michigan
Mental Health Research Institute
205 Washtenav Place
Ann Arbor, MI 48109

Dr. Stephen Kosslyn
Harvard University
1236 William James Hall
33 Kirkland St.
Cambridge, MA 02138

Dr. David R. Lambert
Naval Ocean Systems Center
Code 441T
271 Catalina Boulevard
San Diego, CA 92152

Dr. Pat Langley
University of California
Department of Information
and Computer Science
Irvine, CA 92717

Dr. Marcy Lansman
University of North Carolina
The L. L. Thurstone Lab.
Davie Hall 013A
Chapel Hill, NC 27514

Dr. Robert Lawler
Information Sciences, FRL
GTE Laboratories, Inc.
40 Sylvan Road
Waltham, MA 02254

Dr. Alan M. Lesgold
Learning R&D Center
University of Pittsburgh
Pittsburgh, PA 15260

Dr. Alan Leshner
Deputy Division Director
Behavioral and Neural Sciences
National Science Foundation
1800 G Street
Washington, DC 20550

Dr. Gary Lynch
University of California
Center for the Neurobiology of
Learning and Memory
Irvine, CA 92717

Dr. Don Lyon
P. O. Box 44
Higley, AZ 85236

Dr. James McBride
Psychological Corporation
c/o Harcourt, Brace,
Javanovich Inc.
1250 West 6th Street
San Diego, CA 92101

Dr. Jay McClelland
Department of Psychology
Carnegie-Mellon University
Pittsburgh, PA 15213

Dr. James L. McGaugh
Center for the Neurobiology
of Learning and Memory
University of California, Irvine
Irvine, CA 92717

Dr. Joe McLachlan
Navy Personnel R&D Center
San Diego, CA 92152

Dr. James McMichael
Assistant for MPT Research,
Development, and Studies
NAVOP 01B7
Washington, DC 20370

Dr. Al Meyrowitz
Office of Naval Research
Code 1133
800 N. Quincy
Arlington, VA 22217-5000

Dr. George A. Miller
Department of Psychology
Green Hall
Princeton University
Princeton, NJ 08540

Dr. Tom Moran
Xerox PARC
3333 Coyote Hill Road
Palo Alto, CA 94304

Chair, Department of
Computer Science
U.S. Naval Academy
Annapolis, MD 21402

Chair, Department of
Systems Engineering
U.S. Naval Academy
Annapolis, MD 21402

Dr. David Navon
Institute for Cognitive Science
University of California
La Jolla, CA 92093

Dr. Allen Newell
Department of Psychology
Carnegie-Mellon University
Schenley Park
Pittsburgh, PA 15213

Dr. Mary Jo Nissen
University of Minnesota
N218 Elliott Hall
Minneapolis, MN 55455

Dr. Donald A. Norman
Institute for Cognitive Science
University of California
La Jolla, CA 92093

Director, Training Laboratory,
NPRDC (Code 05)
San Diego, CA 92152

Director, Manpower and Personnel
Laboratory,
NPRDC (Code 06)
San Diego, CA 92152

Director, Human Factors
& Organizational Systems Lab,
NPRDC (Code 07)
San Diego, CA 92152

Fleet Support Office,
NPRDC (Code 301)
San Diego, CA 92152

Library, NPRDC
Code P201L
San Diego, CA 92152

Dr. Stellan Ohlsson
Learning R & D Center
University of Pittsburgh
3939 O'Hara Street
Pittsburgh, PA 15213

Office of Naval Research,
Code 1133
800 N. Quincy Street
Arlington, VA 22217-5000

Office of Naval Research,
Code 1141NP
800 N. Quincy Street
Arlington, VA 22217-5000

Office of Naval Research,
Code 1142EP
800 N. Quincy Street
Arlington, VA 22217-5000

Office of Naval Research,
Code 1142PT
800 N. Quincy Street
Arlington, VA 22217-5000
(6 Copies)

Psychologist
Office of Naval Research
Branch Office, London
Box 39
FPO New York, NY 09510

Special Assistant for Marine
Corps Matters,
ONR Code 00MC
800 N. Quincy St.
Arlington, VA 22217-5000

Psychologist
Office of Naval Research
Liaison Office, Far East
APO San Francisco, CA 96503

Dr. Judith Orasanu
Army Research Institute
5001 Eisenhover Avenue
Alexandria, VA 22333

Daira Paulson
Code 52 - Training Systems
Navy Personnel R&D Center
San Diego, CA 92152

Dr. James W. Pellegrino
University of California,
Santa Barbara
Department of Psychology
Santa Barbara, CA 93106

Department of Computer Science,
Naval Postgraduate School
Monterey, CA 93940

Dr. Steven Pinker
Department of Psychology
E10-018
M.I.T.
Cambridge, MA 02139

Dr. Martha Polson
Department of Psychology
Campus Box 346
University of Colorado
Boulder, CO 80309

Dr. Peter Polson
University of Colorado
Department of Psychology
Boulder, CO 80309

Dr. Mike Posner
University of Oregon
Department of Psychology
Eugene, OR 97403

Dr. Karl Pribram
Stanford University
Department of Psychology
Bldg. 4201 -- Jordan Hall
Stanford, CA 94305

Dr. Lynne Reder
Department of Psychology
Carnegie-Mellon University
Schenley Park
Pittsburgh, PA 15213

Dr. James A. Reggia
University of Maryland
School of Medicine
Department of Neurology
22 South Greene Street
Baltimore, MD 21201

Dr. Daniel Reisberg
Department of Psychology
New School for Social Research
65 Fifth Avenue
New York, NY 10003

Dr. Gil Ricard
Mail Stop C04-14
Grumman Aerospace Corp.
Bethpage, NY 11714

Dr. David Rumelhart
Center for Human
Information Processing
Univ. of California
La Jolla, CA 92093

Dr. E. L. Saltzman
Haskins Laboratories
270 Crown Street
New Haven, CT 06510

Dr. Arthur Samuel
Yale University
Department of Psychology
Box 11A, Yale Station
New Haven, CT 06520

Dr. Robert Sasmor
Army Research Institute
5001 Eisenhover Avenue
Alexandria, VA 22333

Dr. Walter Schneider
Learning R&D Center
University of Pittsburgh
3939 O'Hara Street
Pittsburgh, PA 15260

Dr. Hans-Willi Schroiff
Institut fuer Psychologie
der RWTH Aachen
Jaegerstrasse zwischen 17 u. 19
5100 Aachen
WEST GERMANY

Dr. Marc Sebrechts
Department of Psychology
Wesleyan University
Middletown, CT 06475

Dr. T. B. Sheridan
Dept. of Mechanical Engineering
MIT
Cambridge, MA 02139

Dr. Randall Shumaker
Naval Research Laboratory
Code 7510
4555 Overlook Avenue, S.W.
Washington, DC 20375-5000

Dr. Miriam Schustack
Code 51
Navy Personnel R & D Center
San Diego, CA 92152

Dr. Herbert A. Simon
Department of Psychology
Carnegie-Mellon University
Schenley Park
Pittsburgh, PA 15213

Dr. Kathryn T. Spoehr
Brown University
Department of Psychology
Providence, RI 02912

Dr. Ted Steinke
Dept. of Geography
University of South Carolina
Columbia, SC 29208

Dr. Saul Sternberg
University of Pennsylvania
Department of Psychology
3815 Walnut Street
Philadelphia, PA 19104

Dr. Paul J. Sticha
Senior Staff Scientist
Training Research Division
HumRRO
1100 S. Washington
Alexandria, VA 22314

Dr. John Tangney
AFOSR/NL
Bolling AFB, DC 20332

Dr. Richard F. Thompson
Stanford University
Department of Psychology
Bldg. 4201 -- Jordan Hall
Stanford, CA 94305

Dr. Michael T. Turvey
Haskins Laboratories
270 Crown Street
New Haven, CT 06510

Dr. James Tweeddale
Technical Director
Navy Personnel R&D Center
San Diego, CA 92152

Headquarters, U. S. Marine Corps
Code MPI-20
Washington, DC 20380

Dr. William Uttal
NOSC, Hawaii Lab
Box 997
Kailua, HI 96734

Dr. Kurt Van Lehn
Department of Psychology
Carnegie-Mellon University
Schenley Park
Pittsburgh, PA 15213

Dr. Beth Warren
Bolt Beranek & Newman, Inc.
50 Moulton Street
Cambridge, MA 02138

Dr. Shih-Sung Wen
Jackson State University
1325 J. R. Lynch Street
Jackson, MS 39217

Dr. Douglas Wetzel
Code 12
Navy Personnel R&D Center
San Diego, CA 92152

Dr. Barry Whitsel
University of North Carolina
Department of Physiology
Medical School
Chapel Hill, NC 27514

Dr. Christopher Wickens
Department of Psychology
University of Illinois
Champaign, IL 61820

Dr. Robert A. Wisher
U.S. Army Institute for the
Behavioral and Social Sciences
5001 Eisenhover Avenue
Alexandria, VA 22333

Dr. Donald Woodward
Office of Naval Research
Code 1141NP
800 North Quincy Street
Arlington, VA 22217-5000

Dr. Wallace Wulfeck, III
Navy Personnel R&D Center
San Diego, CA 92152

Dr. Joe Yasatuke
AFHRL/LRT
Lowry AFB, CO 80230

Mr. Carl York
System Development Foundation
181 Lytton Avenue
Suite 210
Palo Alto, CA 94301

Dr. Joseph L. Young
Memory & Cognitive
Processes
National Science Foundation
Washington, DC 20550

Dr. Steven Zornetzer
Office of Naval Research
Code 1140
800 N. Quincy St.
Arlington, VA 22217-5000

Dr. Michael J. Zyda
Naval Postgraduate School
Code 52CK
Monterey, CA 93943